On the Average Density and Selectivity of Nodes in Multi-Digit Tries

Yuriy A. Reznik *
RealNetworks, Inc.
2601 Elliott Avenue, Suite 1000
Seattle, WA 98121
yreznik@acm.org

Abstract

We introduce and study two parameters of nodes in tries with multi-digit branching. The first parameter, which we call a *density* of a multi-digit node, is a ratio of the number of non-empty pointers (i.e. pointers to the attached non-empty sub-tries) to the total number of pointers in this node. The second parameter, which we call a *selectivity* of a node, is a ratio of the number of pointers to external nodes (containing uniquely identified strings) to the total number of strings processed by this node. We show, that in a memoryless model, the average density and the average selectivity of an r-digit node over n binary strings both yield asymptotic expressions in the form $\Phi\left((r-r^*)\,/\,\left(\sigma^*\sqrt{\log n}\right)\right)$, where, however, their central values r^* and factors σ^* are different if the source is asymmetric. We use our findings to explain several interesting facts in the average behaviour of multi-digit tries, and complement our presentation with a number of experimental results.

1 Introduction

Digital trees (also known as *radix search trees*, or *tries*) represent a convenient way of organizing alphanumeric sequences of variable length that facilitates their fast retrieving, searching, and sorting (cf. [12, 17, 23]). In its simplest form, a trie over a set of n strings from an alphabet containing m symbols, is an m-ary tree, in which each input string corresponds to a unique path (see Fig.1.a). Typical applications of tries include searching and sorting algorithms, exact and approximate string matching, data compression schemes, and so on.

It is well known, that the *average depth* of a trie, which is commonly used to estimate the *average time of a successful search*, is asymptotically $\log n/h + O(1)$, where h is the entropy of a stochastic process used to produce n input strings (cf. [17, 6, 10, 13, 21, 15, 25, 14]). The *average number of nodes* in a trie, which is commonly used to estimate its *size*, is asymptotically $n \log e/h + O(1)$. These estimates are known to be correct for a rather large class of stochastic processes, including *memoryless*, *Markovian*, and ψ-mixed models (cf. [21, 14, 24]).

In an effort to reduce the search time, several modifications of the trie structure have been proposed. For example, a *multi-digit trie* (we use this notation after [2]) is a trie, which processes some constant number of symbols $r > 1$ in each node (see Fig.1.b). It is easy to observe that this modification is r-times faster than a regular (single-digit) trie. However, such an improvement comes at a cost of about m^r/r-times more memory, since $r - digit$ nodes must have m^r pointers, most of which are wasted if r is large.

This motivated the development of *adaptive multi-digit trie* structures (cf. [2, 19, 22]), in which the parameter r (the number of digits to be processed) can be changed from one node to another (see Fig.1.c-d).

The best known example of such a structure is a *level-compressed trie* (or *LC- trie*) of Andersson and Nilsson [2], which simply combines all complete levels of the corresponding m-ary trie. It has been shown [20, 3] that in a memoryless model, it creates nodes with $r \to -\log(n)/\log p_{\min}$, where n is the number of strings processed by a node, and $p_{\min} = \min\{p_i\}$, and p_i $(1 \leqslant i \leqslant m)$ are the probabilities of symbols produced by the source. When the memoryless source is symmetric ($p_i = 1/m$), the expected depth of an LC-trie is only $\sim \log^* n$ [2, 7], however, it grows as $O(\log\log n)$ in the asymmetric case [3].

Nilsson and Tikkanen have recently proposed a modification of an LC-trie [19] which combines all successive levels of the corresponding m-ary trie until they reach one that is 50% full (i.e. they allow up to 50% of pointers to be empty). While it was shown that such an algorithm works substantially faster than the original $LC-trie$ in practice [19], the rigorous analysis of its asymptotic behavior has not been provided yet. It is not known, for example, how large such tree can

*On leave from the Institute of Mathematical Machines and Systems, Kiev, Ukraine.

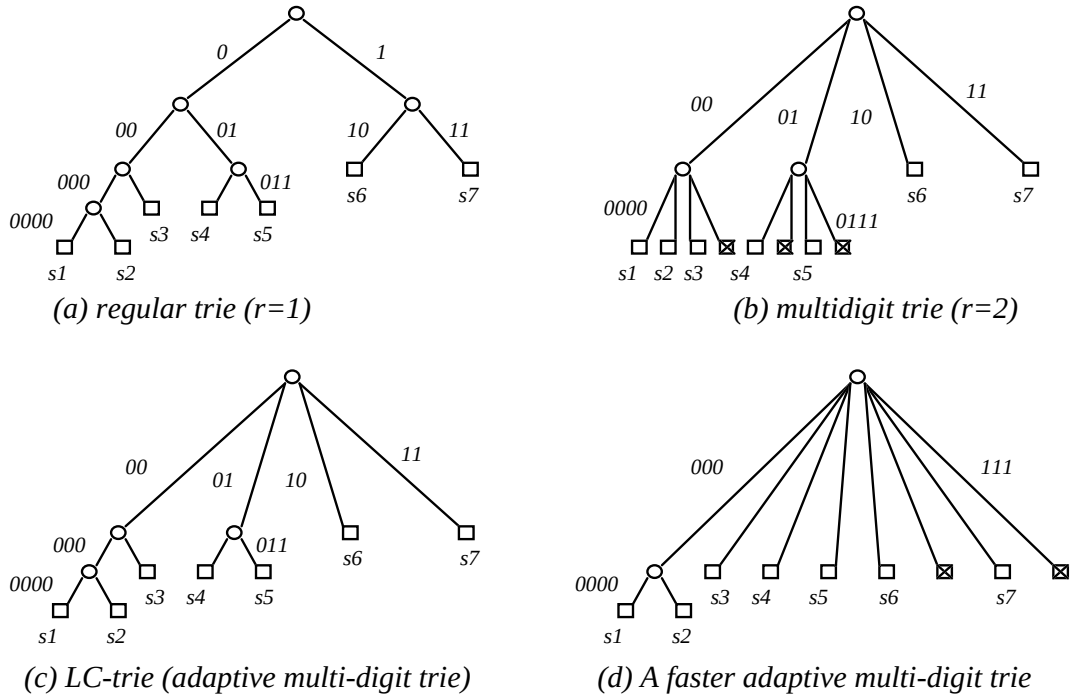

Figure 1: Examples of tries built from 7 binary strings: *s1=0000...*, *s2=0001...*, *s3=0010...*, *s4=0100...*, *s5=0110...*, *s6=100...*, *s7=110...*.

be under asymmetric sources, or whether it can attain $O(1)$ speed (at least in a symmetric case). It is not clear also if there exists a heuristic for selecting degrees of nodes (parameters r) leading to a better asymptotic behavior (in time- and/or space- domains).

In this paper, we will attempt to shed some additional light on this problem by introducing and studying two new parameters of nodes in adaptive multi-digit tries. We will show, that a *density* of a multi-digit node, which is a ratio of the number of pointers to its child nodes to the total number of pointers in the node, directly relates to the overall space-efficiency of a trie built from nodes with similar densities. For instance, by studying the average densities of r-digit nodes, we automatically obtain an answer for space-efficiency of "relaxed" LC-tries of Nilsson and Tikkanen [19].

We will also show that a *selectivity* of a multi-digit node, which is a ratio of the number of pointers to strings uniquely identified (selected) by this node to all strings it processes, directly relates to the overall time-efficiency of a trie built from nodes with similar selectivities. This observation will allow us to formulate an algorithm for construction of adaptive tries with constant average depth (existence of such tries was predicted in [22]).

For both parameters (density and selectivity) we will derive their exact and asymptotic expressions in memoryless model, and will discuss several implications of these formulas for the average behavior of adaptive multi-digit tries.

This paper is organized as follows. In the next section, we will give formal definitions, present our main results, and discuss their consequences. The proofs are delayed until Section 3, where we will also briefly describe the required techniques of the asymptotic analysis. Finally, in Section 4, we will show how our parameters (density and selectivity of nodes) can be used for construction of adaptive multi-digit tries, and will present the results of an experimental evaluation of the resulting data structures.

2 Definitions and Main Results

Consider a set of n distinct strings $S = \{s_1, \ldots, s_n\}$, where each string is a sequence of symbols from a binary alphabet $\Sigma = \{0, 1\}$[1]. By s_j^k we denote a suffix of a string s_j that starts at k-th position in this string (i.e.

[1] We use binary alphabet for the simplicity of presentation only. All our results should remain correct (with the appropriate reformulations of constants and bases of logarithms) for any finite alphabet

there exists a string x_j, of length $|x_j| = k$, such that $s_j = x_j\, s_j^k$). By $\mathrm{Bin}_k(i)$ we denote k least significant digits in binary representation of a number i.

A recursive construction of a binary trie over S can be done based on the following definition.

DEFINITION 1. *A* binary trie $T(S)$ *over a set of strings S has the following properties. If $n = 0$, the trie is* empty. *If $n = 1$ (i.e. S has only one string), the trie is an* external node *containing a pointer to this single string in S. If $n > 1$, the trie is an* internal node *containing pointers to* 2 *child tries: $T(S_0)$ and $T(S_1)$, which contain suffixes of strings from S that begin with symbols* 0 *and* 1 *correspondingly $S_i = \{s_j^1 \mid i\, s_j^1 = s_j \in S\}$.*

We depict a trie over a set of 7 binary strings in Fig.1.a. Observe, that all input strings $\{s_1, \ldots, s_7\}$ inserted in a trie can be uniquely identified by the paths from the root node to the corresponding external nodes.

The next two definitions describe the key properties of the fixed-order and adaptive multi-digit tries (the corresponding examples are shown in Fig.1.b and Fig.1.c).

DEFINITION 2. *A* multi-digit trie $T(S)$ *over a set of strings S has the following properties. If $n = 0$, the trie is empty. If $n = 1$, the trie is an external node containing a pointer to a single string in S. If $n > 1$, the trie is an* r-digit internal node ($r \geqslant 1$ *is a given parameter*) *containing pointers to 2^r child tries: $T(S_0), \ldots, T(S_{2^r-1})$, which contain suffixes of strings from S that begin with the corresponding r-digit sequences $S_i = \{s_j^r \mid \mathrm{Bin}_r(i)\, s_j^r = s_j \in S\}$ $(0 \leqslant i < 2^r)$.*

DEFINITION 3. *An* adaptive multi-digit trie *is a multi-digit trie, such that parameters r defining the number of digits processed by its nodes are chosen adaptively from one node to another.*

In this paper, we will be dealing with the following parameters in a trie built over n strings:

A_n the number of internal nodes;

X_n the number of external nodes: $X_n := n$;

E_n the number of empty pointers (i.e. pointers to empty sub-tries);

S_n the total number of pointers in a trie:

$$S_n = A_n - 1 + X_n + E_n; \tag{2.1}$$

C_n the external path length (i.e. the sum of lengths of paths from the root to all external nodes);

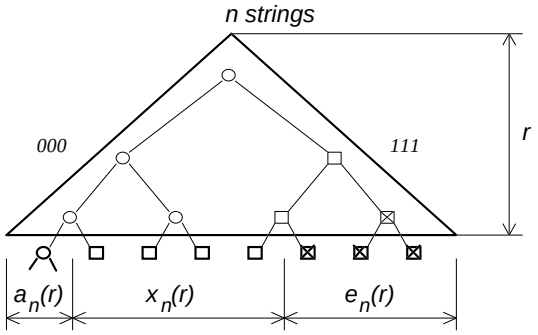

Figure 2: Parameters of an r-digit node processing n strings: $e_n(r)$ - the number of empty pointers, $x_n(r)$ - the number of external nodes, and $a_n(r)$ - the number of the attached internal nodes.

D_n the average depth of a trie:

$$D_n := C_n/n. \tag{2.2}$$

As we mentioned earlier, the average depth D_n of a trie is commonly used to estimate the average *time of a successful search* in a trie. To estimate the *amount of memory* used by a trie we can use the total number of its pointers S_n[2].

Instead of studying these parameters directly, however, in this paper *we will focus our attention on the statistics of their component quantities in multi-digit nodes*. We illustrate our approach in Fig.2.

Consider an r-digit node processing n strings (i.e. we picked a root node of a trie over n strings). We have the following parameters:

$a_n(r)$ the number of pointers to internal nodes attached to this node;

$x_n(r)$ the number of pointers to external nodes (strings);

$e_n(r)$ we denote the number of empty pointers (i.e. pointers to empty child tries).

Observe that the total number of pointers in such a node is

$$e_n(r) + x_n(r) + a_n(r) = 2^r. \tag{2.3}$$

Now we can introduce two quantities of our main interest.

[2]Note that this metric is different from one used in the analysis of regular tries(cf. [17, 10]). In that case, it was sufficient to use the number of internal nodes A_n. However, since the internal nodes in adaptive tries have different sizes, we have to use another parameter (S_n) to take into account these differences as well.

DEFINITION 4. *A* density *of an r-digit node processing n strings $\rho_n(r)$ is a ratio of the number of pointers to all non-empty nodes to the total number of pointers in this node:*

$$\rho_n(r) = \frac{x_n(r) + a_n(r)}{2^r} = 1 - \frac{e_n(r)}{2^r}. \tag{2.4}$$

DEFINITION 5. *A* selectivity *of an r-digit node processing n strings $\xi_n(r)$ is a ratio of the number of strings uniquely identified by this node (i.e. strings referenced by the immediately attached external nodes) to the total number of strings processed by this node:*

$$\xi_n(r) = \frac{x_n(r)}{n}. \tag{2.5}$$

The relation between these parameters of nodes and the characteristics of the entire trie is established by the next two observations.

DEFINITION 6. *A* density *of a multi-digit trie over n strings P_n is a ratio of the number of non-empty pointers to the total number of pointers in the trie:*

$$P_n = \frac{A_n - 1 + X_n}{S_n}; \tag{2.6}$$

OBSERVATION 1. *Consider a multi-digit trie over n strings. Let also*

$$\rho_{\min} = \min_j \left\{\rho_{n_j}(r_j)\right\}, \tag{2.7}$$

$$\rho_{\max} = \max_j \left\{\rho_{n_j}(r_j)\right\}, \tag{2.8}$$

where j enumerates all internal nodes in the trie. Then the density of this trie satisfies:

$$\rho_{\min} \leqslant P_n \leqslant \rho_{\max}. \tag{2.9}$$

OBSERVATION 2. *Consider a multi-digit trie over n strings. Let also*

$$\xi_{\min} = \min_j \left\{\xi_{n_j}(r_j)\right\}, \tag{2.10}$$

$$\xi_{\max} = \max_j \left\{\xi_{n_j}(r_j)\right\}, \tag{2.11}$$

where j enumerates all internal nodes in the trie. Then the average depth D_n of this trie satisfies:

$$\frac{1}{\xi_{\max}} \leqslant D_n \leqslant \frac{1}{\xi_{\min}}. \tag{2.12}$$

In other words, if we know the densities $\rho_{n_j}(r_j)$ and selectivities $\xi_{n_j}(r_j)$ of nodes in a trie, then we automatically obtain upper and lower bounds for the corresponding characteristics of the entire trie.

These parameters are especially useful if we study adaptive tries that naturally impose some constrains on $\rho_{n_j}(r_j)$ or $\xi_{n_j}(r_j)$ during their construction. In fact, we already know several examples of *density constrained* tries in the form of LC-tries and their variants [2, 19]. On the other hand, the use of selectivity for construction of tries has not been explored yet, and in Section 4, we will introduce a new algorithm, constructing *selectivity constrained* tries.

In order to study the *average behaviour* of $\rho_n(r)$ and $\xi_n(r)$ we will assume that our input strings S are generated by a binary *memoryless* (or *Bernoulli*) source [4]. In this model, symbols of the alphabet $\Sigma = \{0, 1\}$ occur independently of one another, so that if x_j is the j-th symbol produced by this source, then for any j: $\Pr\{x_j = 0\} = p$, and $\Pr\{x_j = 1\} = q = 1 - p$. If $p = q = 0.5$, such source is called *symmetric*, otherwise it is *asymmetric*.

Now, we can express:

$$\bar{\rho}_n(r) := E\{\rho_n(r)\} = 1 - \frac{E\{e_n(r)\}}{2^r}, \tag{2.13}$$

$$\bar{\xi}_n(r) := E\{\xi_n(r)\} = \frac{E\{x_n(r)\}}{n}, \tag{2.14}$$

where expectations are taken over all possible tries over n strings when parameters of the memoryless source (p and q) are fixed.

We are now ready to present our main results regarding the expected densities and selectivities of nodes in multi-digit tries.

THEOREM 2.1. *The expected density $\bar{\rho}_n(r)$ of an r-digit node processing n binary strings from a memoryless source is:*

$$\bar{\rho}_n(r) = 1 - 2^{-r} \sum_{s=0}^{r} \binom{s}{r} \left(1 - p^s q^{r-s}\right)^n. \tag{2.15}$$

If $p \neq q$ and

$$r = \frac{\log n}{h_\rho} + x\sigma_\rho \sqrt{\log n}, \tag{2.16}$$

where

$$h_\rho = -\log\sqrt{pq} = -\frac{1}{2}\log p - \frac{1}{2}\log q, \tag{2.17}$$

$$h_\rho^{(2)} = \frac{1}{2}\log^2 p + \frac{1}{2}\log^2 q,$$

$$\sigma_\rho^2 = \frac{h_\rho^{(2)} - h_\rho^2}{h_\rho^3}, \tag{2.18}$$

and $x = O(1)$, then, asymptotically, with $n \to \infty$:

$$\bar{\rho}_n(r) = \Phi(-x)\left(1 + O\left(\frac{1}{\sqrt{\log n}}\right)\right), \tag{2.19}$$

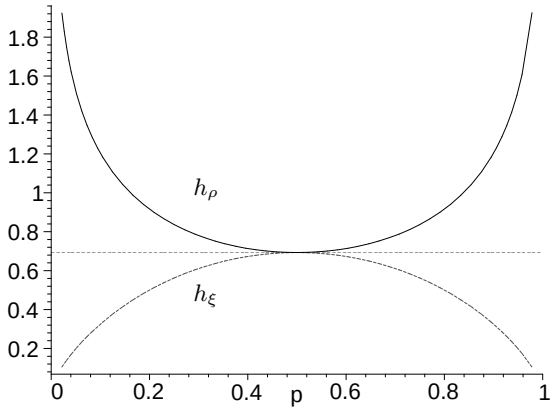

Figure 3: Dependencies of the parameters h_ρ (solid line) and h_ξ (dashed line) on the probabilities p, $q = 1 - p$ of the source.

where

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{t^2}{2}} dt, \tag{2.20}$$

is the distribution function of the standard normal distribution [1].

THEOREM 2.2. *The expected selectivity $\bar{\xi}_n(r)$ of an r-digit node processing n binary strings from a memoryless source is:*

$$\bar{\xi}_n(r) = \sum_{s=0}^{r} \binom{s}{r} p^s q^{r-s} \left(1 - p^s q^{r-s}\right)^{n-1}. \tag{2.21}$$

If $p \neq q$ and

$$r = \frac{\log n}{h_\xi} + x\sigma_\xi \sqrt{\log n}, \tag{2.22}$$

where

$$\begin{aligned} h_\xi &= -p\log p - q\log q, \\ h_\xi^{(2)} &= p\log^2 p + q\log^2 q, \end{aligned} \tag{2.23}$$

$$\sigma_\xi^2 = \frac{h_\xi^{(2)} - h_\xi^2}{h_\xi^3}, \tag{2.24}$$

and $x = O(1)$, then, asymptotically, with $n \to \infty$:

$$\bar{\xi}_n(r) = \Phi(x)\left(1 + O\left(\frac{1}{\sqrt{\log n}}\right)\right). \tag{2.25}$$

where $\Phi(x)$ is the distribution function of the standard normal distribution (2.20).

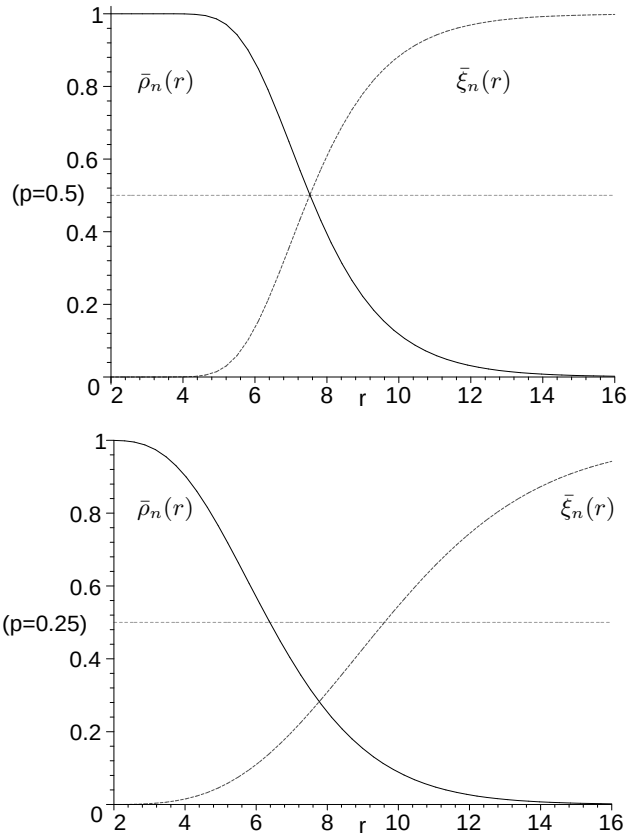

Figure 4: Plots of the average density $\bar{\rho}_n(r)$ (solid line) and the average selectivity $\bar{\xi}_n(r)$ (dashed line) of an r-digit node processing $n = 128$ strings in symmetric ($p = q = 0.5$) and asymmetric ($p = 0.25$) cases.

Observe, that the key parameters h_ρ and h_ξ defining the middle-points in the asymptotic expressions (2.19,2.25)for $\bar{\rho}_n(r)$ and $\bar{\xi}_n(r)$ are not always the same:

$$h_\xi \leqslant \log 2 \leqslant h_\rho. \tag{2.26}$$

The equality in (2.26) is attained only if the memoryless source is symmetric $p = q = 1/2$. However, if the source is asymmetric, the difference between h_ρ and h_ξ can be arbitrary large (see Fig.3).

Consequently, the values of parameters r necessary to keep $\bar{\rho}_n(r)$ and $\bar{\xi}_n(r)$ within certain ranges (e.g. $[0.25, 0.5]$) will be different if the source is asymmetric (see Fig.4).

This explains why, for example, a multi-digit trie that attempts to keep nodes within a certain range of densities, may not be able to control their selectivities (and consequently, the constant depth of the trie). On the other hand, a trie that attempts to keep nodes within a certain range of selectivities, may not be able to

control their densities (and consequently, the constant density (and linear size) of the trie).

These observations may also suggest, that from practical standpoint, it may make most sense to design tries that represent some compromise between these two strategies. For example, we can choose r such that $\bar{\rho}_n(r) = \bar{\xi}_n(r)$ (a point of intersection of curves $\bar{\rho}_n(r)$ and $\bar{\xi}_n(r)$ on Fig.4), which asymptotically leads to $r = \log_2 n + O(\log\log n)$. We conjecture that such a trie is $O(\log\log n)$-fast and $O(n \log\log n)$-large under an asymmetric source, but a detailed analysis is left to a subsequent paper.

We will give formal definitions of several algorithms for construction of adaptive multi-digit tries based on density and selectivity constrains in Section 4, where we will also evaluate them experimentally.

3 Analysis

To explain claims (2.9) and (2.12) of our Observations 1 and 2 we will use the following simple fact.

LEMMA 3.1. *Let x_1, y_1, and x_2, y_2 be some positive real numbers, and:*

$$\frac{y_1}{x_1} \leqslant \frac{y_2}{x_2}. \tag{3.27}$$

Then:

$$\frac{y_1}{x_1} \leqslant \frac{y_1+y_2}{x_1+x_2} \leqslant \frac{y_2}{x_2}. \tag{3.28}$$

Proof. Observe, that condition (3.27) also implies that

$$\frac{y_1}{y_2} \leqslant \frac{x_1}{x_2}. \tag{3.29}$$

Then:

$$\frac{y_1+y_2}{x_1+x_2} = \frac{y_2}{x_2}\left(\frac{1+y_1/y_2}{1+x_1/x_2}\right) \leqslant \frac{y_2}{x_2},$$

where the last transition is due to (3.29). The left side of (3.28) is proved in essentially the same way.

Applying (3.28) recursively, we can show that for an arbitrary set of pairs of positive numbers $\{x_i, y_i\}$:

$$\min_i \frac{y_i}{x_i} \leqslant \frac{\sum_i y_i}{\sum_i x_i} \leqslant \max_i \frac{y_i}{x_i}.$$

It remains to notice that:

$$P_n = \frac{A_n - 1 + X_n}{S_n} = \frac{\sum_i \left(a_{n_i}(r_i) + x_{n_i}(r_i)\right)}{\sum_i 2^{r_i}},$$

and

$$1/D_n = \Xi_n = \frac{X_n}{C_n} = \frac{\sum_i x_{n_i}(r_i)}{\sum_i n_i},$$

where n_i, r_i, $a_{n_i}(r_i)$, and $x_{n_i}(r_i)$ are the corresponding parameters of nodes of a trie.

For the purpose of compact presentation of proofs of formulas (2.15) and (2.21) in our Theorems 1 and 2, we will introduce the following parameter.

DEFINITION 7. *Consider an r-digit node processing n binary strings. By $z_n^k(r)$ we denote the number of its child nodes that contain exactly k strings from the original set of n.*

We immediately notice, that all the previously defined parameters of r-digit nodes can be easily obtained using $z_n^k(r)$:

$$e_n(r) = z_n^0(r), \tag{3.30}$$

$$x_n(r) = z_n^1(r), \tag{3.31}$$

$$a_n(r) = \sum_{k=2}^{n} z_n^k(r) = 2^r - z_n^0(r) - z_n^1(r). \tag{3.32}$$

The next lemma provides an exact formula for the average value of $z_n^k(r)$ in a memoryless model.

LEMMA 3.2. *The quantity $\bar{z}_n^k(r) := E\left\{z_n^k(r)\right\}$ in a memoryless model satisfies:*

$$\bar{z}_n^k(r) = \binom{n}{k} \sum_{s=0}^{r} \binom{r}{s} \left(p^s q^{r-s}\right)^k \left(1 - p^s q^{r-s}\right)^{n-k}. \tag{3.33}$$

Proof. Consider an r-digit node processing n strings. Assuming that each of its 2^r branches have probabilities $p_1, \ldots, p_{2^r}$, and using the standard technique for enumeration of nodes in tries [17, 6.3-3], we can write:

$$\begin{aligned}
\bar{z}_n^k &= \sum_{l_1 \ldots l_{2^r}} \binom{n}{l_1 \ldots l_{2^r}} p_1 \ldots p_{2^r} \left(\delta_{kl_1} + \ldots + \delta_{kl_{2^r}}\right) \\
&= \sum_{l=0}^{n} \binom{n}{l} \left(p_1^l (1-p_1)^{n-l} + \ldots + p_{2^r}^l (1-p_{2^r})^{n-l}\right) \delta_{kl} \\
&= \binom{n}{k} \left(p_1^k (1-p_1)^{n-k} + \ldots + p_{2^r}^k (1-p_{2^r})^{n-k}\right),
\end{aligned} \tag{3.34}$$

where δ_{ij} is a Kronecker delta. Recall now, that we are actually working with an r-digit node, so given the probabilities of each digit (p and $q = 1-p$ for symbols 0 and 1 correspondingly) we can write:

$$p_i = p^{s_i} q^{r-s_i}, \tag{3.35}$$

where s_i is the number of occurrences of symbol 0 in a string leading to a branch i ($1 \leqslant i \leqslant 2^r$). Combining (3.34) and (3.35), we arrive at the expression (3.33) claimed by the lemma.

Using the result of this lemma (3.33), mappings (3.30,3.31), and formulas for the average density (2.13) and the average selectivity of an r-digit node we can show that:

$$\begin{aligned}\bar{\rho}_n(r) &= 1-\frac{\bar{z}_n^0}{2^r} = 1-2^{-r}\sum_{s=0}^{r}\binom{r}{s}\left(1-p^s q^{r-s}\right)^n,\\ \bar{\xi}_n(r) &= \frac{\bar{z}_n^1}{n} = \sum_{s=0}^{r}\binom{r}{s}p^s q^{r-s}\left(1-p^s q^{r-s}\right)^{n-1},\end{aligned}$$

which are exactly the expressions (2.15) and (2.21) in our Theorems.

In order to evaluate asymptotic behaviours of (2.15) and (2.21) for large n, we will convert them into alternating sums:

$$\bar{\rho}_n(r) = 1-2^{-r}\sum_{k=0}^{n}\binom{n}{k}(-1)^k\left(p^k+q^k\right)^r, \tag{3.36}$$

$$\bar{\xi}_n(r) = \sum_{k=0}^{n-1}\binom{n-1}{k}(-1)^k\left(p^{k+1}+q^{k+1}\right)^r, \tag{3.37}$$

and apply Rice's integral method (cf. Knuth [17, Ex.5.2.2-54], Flajolet and Sedgewick [10, 11]).

We quote the following formulation of this method from [24].

LEMMA 3.3. (S.O.RICE) *Let $f(z)$ be of polynomial growth at infinity, and analytical left to the vertical line $\left(\frac{1}{2}-m-i\infty,\ \frac{1}{2}-m+i\infty\right)$. Then*

$$\begin{aligned}&\sum_{k=m}^{n}\binom{n}{k}(-1)^k f(k)\\ &= \frac{1}{2\pi i}\int_{\frac{1}{2}-m-i\infty}^{\frac{1}{2}-m+i\infty} f(-z)B(n+1,z)dz\\ &= \frac{1}{2\pi i}\int_{\frac{1}{2}-m-i\infty}^{\frac{1}{2}-m+i\infty} f(-z)n^{-z}\Gamma(z)\left(1+O\left(\frac{1}{n}\right)\right)dz,\end{aligned}$$

where $B(x,y)=\Gamma(x)\Gamma(y)/\Gamma(x+y)$ is the beta-function.

Using this result in the alternating sum for the expected density of an r-digit node (3.36), we obtain

$$\begin{aligned}\bar{\rho}_n(r) &= 1-\left(1+O\left(\frac{1}{n}\right)\right)\times\\ &\times\frac{1}{2\pi i}\int_{\frac{1}{2}-i\infty}^{\frac{1}{2}+i\infty} n^{-z}\Gamma(z)\,2^{-r}\left(p^{-z}+q^{-z}\right)^r dz.\end{aligned} \tag{3.38}$$

Observe, that a function under the integral (3.38) has a saddle point coinciding with a pole ($z=0$) of $\Gamma(z)$ (see Fig.5). Hence, we have to use the *saddle point method* [5]. Due to the space constrains we will only sketch the main steps in this process, and a rigorous proof (which requires the use of Van der Waerden technique [26]) is delayed till the final version of this paper.

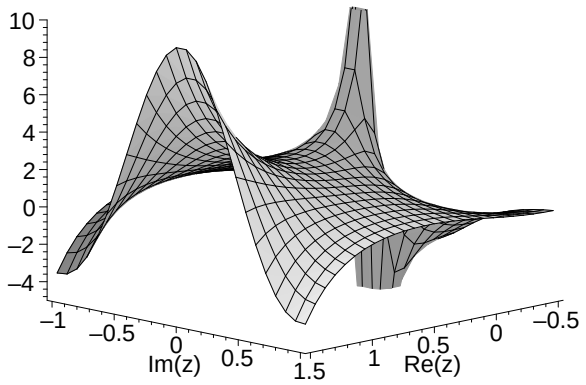

Figure 5: A saddle point of $n^{-z}\Gamma(z)\,2^{-r}\left(p^{-z}+q^{-z}\right)^r$ near the pole ($z=0$) of $\Gamma(z)$. The plot is rendered for $n=1024$, $r=10$, and $p=0.325$.

To apply the saddle point method we write:

$$\begin{aligned}&n^{-z}2^{-r}\left(p^{-z}+q^{-z}\right)^r\\ &= e^{-\left(z\log n+r\left(\log 2-\log\left(p^{-z}+q^{-z}\right)\right)\right)}\\ &= e^{-\left(\frac{\left(\log n-rh_\rho\right)^2}{2r\left(h_\rho^{(2)}-h_\rho^2\right)}-r\frac{h_\rho^{(2)}-h_\rho^2}{2}(z-s_0)^2+O\left(r\,z^3\right)\right)},\end{aligned} \tag{3.39}$$

where

$$s_0 = \frac{\log n - rh_\rho}{r\left(h_\rho^{(2)}-h_\rho^2\right)} = \frac{-x}{h_\rho\sigma_\rho\sqrt{\log n}}+O\left(\frac{x^2}{\log n}\right), \tag{3.40}$$

and h_ρ, $h_\rho^{(2)}$, σ_ρ, and x are as defined in (2.16-2.18).

Let now $z=s_0+it$ ($-\infty<t<\infty$). Since s_0 is close to the pole of $\Gamma(z)$ at zero (3.40), we must use (at least, principal part of) its Laurent series:

$$\Gamma(z)\sim\frac{1}{s_0+it}=\frac{s_0}{s_0^2+t^2}-i\frac{t}{s_0^2+t^2}.$$

Combining the above formulas and substituting $u=t^2$ we arrive at

$$\begin{aligned}&\frac{1}{\pi i}\int_0^\infty e^{-\left(\frac{x^2}{2}+\frac{1}{2}\left(\log n+xh_\rho\sigma_\rho\sqrt{\log n}\right)h_\rho^2\sigma_\rho^2t^2+O\left(\frac{x^3}{\sqrt{\log n}}\right)\right)}\frac{s_0\,dt}{s_0^2+t^2}\\ &= \left(1+O\left(\frac{x^3}{\sqrt{\log n}}\right)\right)\frac{s_0}{\pi i}\,e^{-\frac{x^2}{2}}\int_0^\infty e^{-u\beta}\frac{1}{\sqrt{u}\,(s_0^2+u)}du,\end{aligned}$$

where

$$\beta=\frac{1}{2}\left(\log n+xh_\rho\sigma_\rho\sqrt{\log n}\right)h_\rho^2\sigma_\rho^2.$$

But, it can be shown that

$$\frac{1}{\pi i}\int_0^\infty e^{-u\beta}\frac{1}{\sqrt{u}\,(s_0^2+u)}du = \frac{1}{2s_0}e^{\beta s_0^2}\mathrm{Erfc}\left(s_0\sqrt{\beta}\right)\ ,$$

where

$$\mathrm{Erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^\infty e^{-t^2}dt\ ,$$

is a complementary error function [1]. Then, by observing that

$$s_0\sqrt{\beta} = \frac{-x}{\sqrt{2}} + O\left(\frac{x^2}{\sqrt{\log n}}\right),$$

and

$$-\frac{x^2}{2} + \beta s_0^2 = O\left(\frac{x^3}{\sqrt{\log n}}\right),$$

we finally obtain

$$\begin{aligned}\bar{\rho}_n(r) &= 1 - \frac{1}{2}\,\mathrm{Erfc}\left(\frac{-x}{\sqrt{2}}\right)\left(1+O\left(\frac{1}{\sqrt{\log n}}\right)\right)\\ &= \Phi(-x)\left(1+O\left(\frac{1}{\sqrt{\log n}}\right)\right),\end{aligned}$$

which is the asymptotic expression (2.19) claimed by the Theorem 1.

The proof of the asymptotic expression for selectivity of an r-digit node (2.25) is obtained in essentially the same way. In fact, the asymptotic behaviour of an alternating sum almost identical to (3.37) has already been studied by Louchard [18] and Szpankowski [24, Ex.8.19].

4 Experimental Results

In this section we will study the following four implementations of adaptive multi-digit tries.

The first algorithm is a well-known level-compressed trie due to Andersson and Nilsson [2].

Definition 8. *A level-compressed trie is an adaptive multi-digit trie, such that the number of digits r processed by its nodes with n inserted strings satisfy:*

$$r = \max\{j \mid \rho_n(j) = 1\}, \tag{4.41}$$

The second algorithm is a density-constrained version a level-compressed trie, proposed by Nilsson and Tikkanen [19].

Definition 9. *A* density-constrained multi-digit trie *is an adaptive multi-digit trie, such that the number of digits r processed by its nodes over n strings satisfy:*

$$r \in \{j \mid \rho_1 \leqslant \rho_n(j) \leqslant \rho_2\}, \tag{4.42}$$

where $0 < \rho_1 \leqslant \rho_2 \leqslant 1$ are some positive constants.

The third algorithm, which (to the best of the author's knowledge) is novel, uses the *selectivity* of a multi-digit node to control the construction.

Definition 10. *A* selectivity-constrained multi-digit trie *is an adaptive multi-digit trie, such that the number of digits r processed by its nodes over n strings satisfy:*

$$r \in \{j \mid \xi_1 \leqslant \xi_n(j) \leqslant \xi_2\}, \tag{4.43}$$

where $0 < \xi_1 \leqslant \xi_2 \leqslant 1$ are some positive constants.

Finally, we also consider an algorithm, which we call a *logarithmic trie*[3], defined as follows.

Definition 11. *A* logarithmic multi-digit trie *is an adaptive multi-digit trie, such that the number of digits r processed by its nodes over n strings satisfy:*

$$r = \lceil \log_2 n \rceil. \tag{4.44}$$

In Fig. 6 and Fig. 7 we present the results of experimental evaluation of the expected successful search time and expected memory usage of the above mentioned implementations of multi-digit tries.

To build our tries we used computer-generated sequences of binary digits for symmetric ($p = 0.5$) and asymmetric ($p = 0.25$) cases. To be able to identify even extremely slowly growing functions (e.g. $\log^* n$) we allowed the number of strings n in tries to grow from 2 to 10^5. At each point (each fixed value for n) we generated 10^3 different tries (using the same source model), and estimated their expected depths and relative sizes.

Observe that in the symmetric case (see Fig.6.a), LC-trie is the only structure which depth is clearly increasing with n (and its rate well matches the theoretic estimate $O(\log^* n)$ [2]). The depths of the other tries are fluctuating in the range $(1.5, 2.5)$ with no visible drift upward, which suggests that they are likely $O(1)$-fast.

The situation is quite different in the asymmetric case (see Fig.6.b). Here, both LC- and 50%-dense tries are growing pretty rapidly (likely at $O(\log\log n)$ rate), logarithmic trie are also growing, but at a somewhat slower rate, while 50%-selective tries just fluctuate in the range $(1.8, 2.8)$, which suggests that they are $O(1)$-fast.

Analyzing the results for the expected relative sizes (see Fig. 7.a), we can conjecture that all of our modifications are $O(n)$-large when the source is symmetric. In the asymmetric case (see Fig.7.b), however, the 50%-selective tries tend to grow very rapidly (at least with $O\left(n^{\log 2/h_\xi}\right)$ rate). At the same time, the relative sizes of both LC- and 50%-dense tries are fluctuating between constants, which indicates that they both are $O(n)$-large.

[3] This algorithm can also be interpreted as a multi-digit implementation of N-trees, cf. Dobosiewitz [8] and Ehrlich [9].

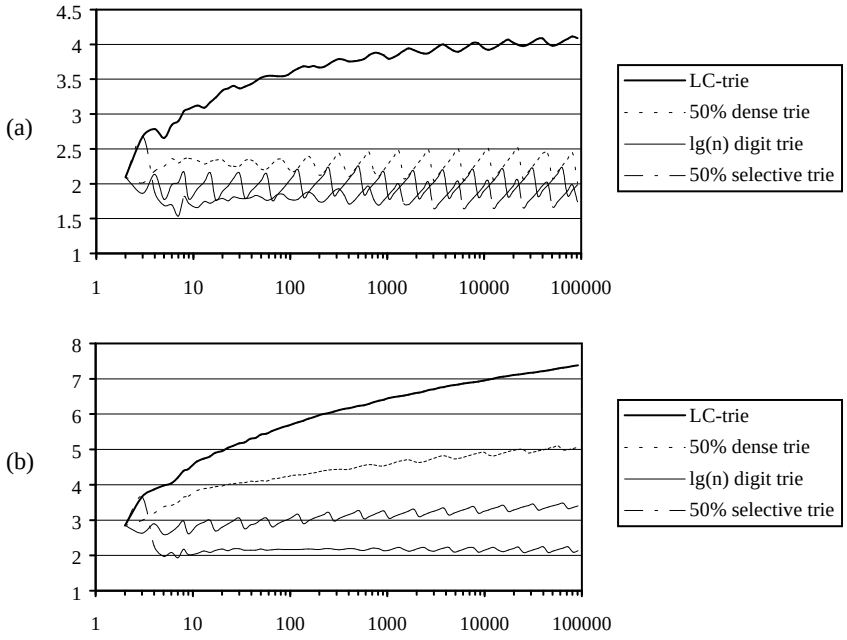

Figure 6: Average depths C_n/n of several modifications of multi-digit tries when (a) memoryless source is symmetric ($p = 0.5$), and (b) the source is asymmetric ($p = 0.25$). Axis x represents the number of strings n inserted in tries.

5 Conclusions

We have introduced and studied parameters of density and selectivity of nodes in adaptive multidigit tries, and have shown how they can be used to construct tries with easily controllable space- and time- characteristics. We have shown, that in the asymmetric memoryless model these strategies lead to principally different asymptotic modes (where with growing asymmetry density-constrained tries become much less efficient in time, and selectivity-constratined tries become prohibitively large in space), and that better time/space compromise is achieved by logarithmic tries (i.e. tries with $\log_2 n$-level nodes).

References

[1] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions*, (Dover, New York, 1972).

[2] A. Andersson and S. Nilsson, Improved Behaviour of Tries by Adaptive Branching, *Information Processing Letters*, **46** (1993) 295–300.

[3] A. Andersson and S. Nilsson, Faster Searching in Tries and Quadtries – An Analysis of Level Compression, *Proc. 2nd Annual European Symp. on Algorithms*(1994) 82–93.

[4] T. M. Cover and J. M. Thomas, *Elements of Information Theory*, (John Wiley & Sons, New York, 1991).

[5] N. G. de Bruijn, *Asymptotic Methods in Analysis* (Dover, NY, 1981)

[6] L. Devroye, A Note on the Average Depths in Tries, *SIAM J. Computing*, **28** (1982) 367–371.

[7] L. Devroye, Analysis of Random LC Tries, *Rand. Structures & Algorithms* **19** (3-4) (2001) 359–375.

[8] W. Dobosiewitz, Sorting by Distributive Partitioning, *Information Processing Letters*, **7** (1) (1978) 1–6.

[9] G. Ehrlich, Searching and Sorting Real Numbers, *J. Algorithms*, **2** (1981) 1–14.

[10] P. Flajolet and R. Sedgewick, Digital Search Trees Revisited, *SIAM J. Computing*, **15** (1986) 748–767.

[11] P. Flajolet and R. Sedgewick, Mellin Transforms and Asymptotics: Finite Differences and Rice's Integrals, *Theoretical Computer Science*, **144** (1995) 101–124.

[12] E. Fredkin, Trie Memory, *Comm. ACM*, **3** (1960) 490–500.

[13] P. Jacquet and M. Régnier, Trie Partitioning Process: Limiting Distributions, *Lecture Notes in Computer Science*, **214** (Springer-Verlag, New York, 1986) 196–210.

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

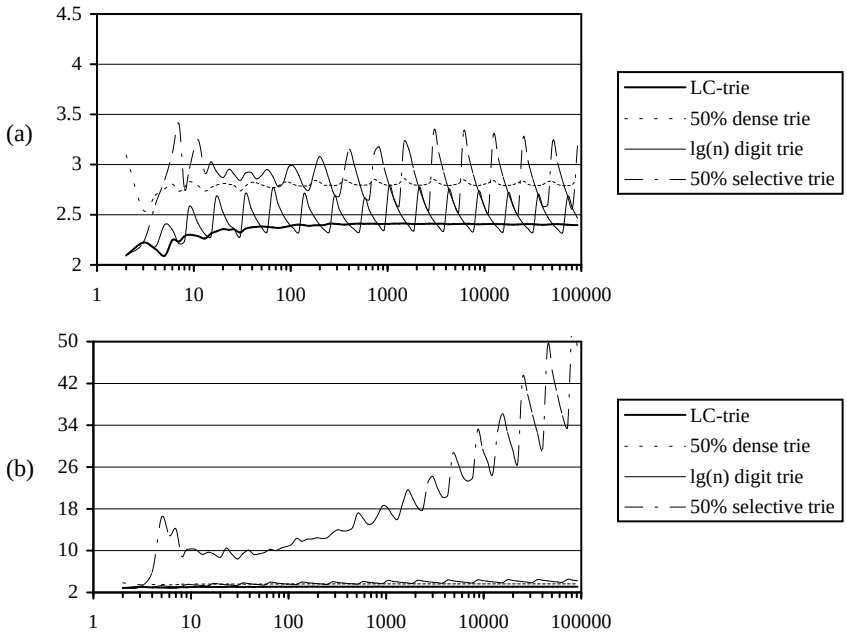

Figure 7: Average relative sizes S_n/n of several modifications of AMD tries when (a) memoryless source is symmetric ($p = 0.5$), and (b) the source is asymmetric ($p = 0.25$). Axis x represents the number of strings n inserted in tries.

[15] P. Kirschenhofer and H. Prodinger, Some further results on digital search trees, *Lecture Notes in Computer Science*, **229** (Springer-Verlag, New York, 1986) 177–185.

[16] D. Knuth, *The Art of Computer Programming. Fundamental Algorithms. Vol. 1* (Addison-Wesley, Reading MA, 1968).

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

[18] G. Louchard, The Brownian Motion: A Neglected Tool for the Complexity Analysis of Sorted Tables Manipulations, RAIRO Theoretical Informatics, **17** (1983) 365–385.

[19] S. Nilsson and M. Tikkanen, Implementing a Dynamic Compressed Trie, *Proc. 2nd Workshop on Algorithm Engineering (WAE'98)*, Saarbruecken, Germany (1998) 25–36.

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

[21] B. Pittel, Paths in a Random Digital Tree: Limiting Distributions, *Advances in Applied Probability*, **18** (1986) 139–155.

[22] Yu. A. Reznik, Some Results on Tries with Adaptive Branching, *Lecture Notes in Computer Science*, **1858** (Springer-Verlag, New York, 2000) 148–158.

[23] R. Sedgewick and P. Flajolet, *An Introduction to the Analysis of Algorithms*, (Addison-Wesley, Reading MA, 1996).

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

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

[26] B. Van der Waerden, On the Method of Saddle Points, *Applied Scientific Research*, **B2** (1951) 33–45