J. Basic. Appl. Sci. Res., 2(9)9275-9281, 2012

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

On Symmetric d-ary Tries: Profile, Depth and Height

Ramin Kazemi

Department of Statistics, Imam Khomeini International University, Qazvin, Iran

ABSTRACT

Tries are fundamental to a number of algorithmic schemes, including radix-based searching and sorting, lossless text compression, dynamic hashing algorithms, communication protocols of the tree or stack type, distributed leader election, and so on. The profile of a trie is a parameter that represents the number of nodes with the same distance to the root. In this paper we obtain the mean and variance of the profile in symmetric d-ary tries. More precisely, we obtain the formulas (explicitly and asymptotically) over strings generated by an extended memoryless source. Also we discuss on the distance from the root to a randomly selected node (depth) and the length of the longest path from the root (height) of the such trees. The results for $d = 2$ reduce to the previous results on symmetric 2-ary tries.
KEY WORDS: Symmetric d-ary tries, profile, depth, height, memoryless source.
2012 Mathematical Subject Classification: 05C05.

1 INTRODUCTION

Tries occur in a variety of computer and communication algorithms including symbolic manipulations, compiling, and comparison based searching and sorting, digital retrieval techniques, algorithms on strings, file systems, codes and communication protocols.

In computer science, a trie, or prefix tree, is an ordered tree data structure that is used to store an associative array where the keys are usually strings. Unlike a search tree, no node in the tree stores the key associated with that node; instead, its position in the tree defines the key with which it is associated. All the descendants of a node have a common prefix of the string associated with that node, and the root is associated with the empty string. Values are normally not associated with every node, only with leaves and some inner nodes that correspond to keys of interest.

Unlike most other algorithms, tries have the peculiar feature that the code path, and hence the time required, is almost identical for insert, delete, and find operations. As a result, for situations where code is inserting, deleting and finding in equal measure, tries can handily beat d-ary search trees, as well as provide a better basis for the CPU's instruction and branch caches.

Tries have been widely studied in the literature (see [15, 20] and the references therein). The motivation of studying of such trees is multifold. First, they are fine shape measures closely connected to many other cost measures on tries; some of them are indicated below. Second, they are also asymptotically close to the profiles of suffix trees, which in turn have a direct combinatorial interpretation in terms of words, and another interpretation in terms of urn models. Third, not only the analytic problems are mathematically challenging, but the diverse new phenomena they exhibit are highly interesting and unusual. Fourth, our findings imply several new results on other shape parameters. Finally, most properties of random tries have also a prototype character and are expected to hold for other varieties of digital search trees (and under more general random models), although the proofs are generally more complicated (see [1, 4, 5, 6, 21]).

2 d-ARY TRIES

Tries are prototype data structures useful for many indexing and retrieval purposes. They were first proposed by de la Briandais [1] in the late 1950's for information processing; Fredkin [10] suggested the current name as it being part of retrieval. Tries are multiway trees whose nodes are vectors of characters or digits. Due to their simplicity and efficiency, tries found widespread use in diverse applications ranging from document taxonomy to IP addresses lookup, from data compression to dynamic hashing, from partial-match queries to speech recognition, from leader election algorithms to distributed hashing tables (see [11, 13, 15, 20]), they are also used to represent the event history in data race detection for multi-threaded object oriented programs (see [2]); The structure of tries also have a close connection to several splitting procedures using coin-flipping; these include algorithms for resolving collisions in multi-access (or broadcast) communication models and algorithms for loser selection or leader election, etc.; see [14]. Thus most shape parameters in tries have direct interpretations in terms of other related objects.

***Corresponding Author:** Ramin Kazemi, Department of Statistics, Imam Khomeini International University, Qazvin, Iran, Email: kazemi@ikiu.ac.ir, Tel: 00982818371377, Fax: 00982813780040

Tries are natural choice of data structures when the input records involve a notion of digits (or alphabets). They are often used to store such data so that future retrieval can be made efficient. Given a sequence of n strings over the d-ary digit $\Sigma = \{0,1,...,i-1,i,i+1,...,d\}$, we can construct a trie as follows. If $n = 0$, then the trie is empty. If $n = 1$, then a single (external) node holding the strings is allocated. If $n \geq d$, then the trie consists of a root (internal) node directing strings to the d subtrees according to the first digit of each string, and strings directed to the same subtree are themselves tries. Unlike other search trees such as digital search trees and d-ary search trees where records or keys are stored at the internal nodes, the internal nodes in tries are branching nodes used merely to direct records to each subtries, records being all stored in external nodes that are leaves of such tries [16]. Figure 1 shows a 2-ary trie built on eight strings $x_1,...,x_8$ (i.e., $x_1 = 0...$, $x_2 = 1...$, $x_3 = 01...$, $x_4 = 11...$, etc.) with internal (ovals) and external (squares) nodes.

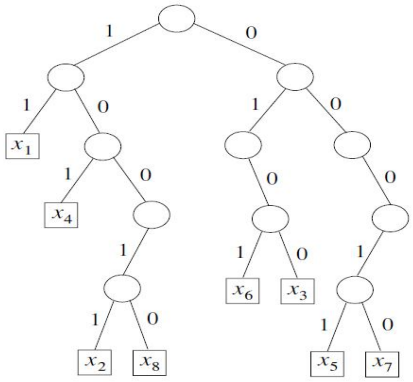

Figure 1: A 2-ary trie built on eight strings $x_1,...,x_8$.

Trie also provides a model for the analysis of several important algorithms, such as Radix Exchange Sort [13], and Extendible Hashing [10].

The profile of a trie is a parameter that represents the number of nodes (either internal or external) with the same distance to the root (the root being at level zero). Such a profile is very informative and closely related to many other shape parameters, although it does not uniquely characterize the tree. It is a function of the number of strings stored in a tree and the distance from the root. Several, if not all, trie parameters such as height, size, depth, shortest path, and fill-up level can be expressed in terms of the (external and internal) profiles [7]. Although profiles represent one of the most fundamental parameters of tries, they have been hardly studied in the past. The analysis of profiles is surprisingly arduous but once it is carried out it reveals unusually intriguing and interesting behavior [6]. We present a detailed study of the distribution of the profiles in a d-ary trie built over strings generated by an extended memoryless source.

Throughout the paper, we write $B_{n,k}$ to denote the number of external nodes (leaves) at distance k from the root; the number of internal nodes at distance k from the root is denoted by $I_{n,k}$ in d-ary tries. For simplicity, we will refer to $B_{n,k}$ as the external profile and $I_{n,k}$ the internal profile. Also we write D_n to denote the distance from the root to a randomly selected node (depth) and H_n to denote the length of the longest path from the root (height) of the such trees.

In this paper we study the above statistics of a d-ary trie built over n d-ary strings generated by an appropriate memoryless source. More precisely, we assume each string is a d-ary i.i.d. sequence with p_i being the probability of a "i" (i=1, ... , d). The corresponding d-ary trie constructed from these n bit-strings is called a random d-ary trie. For the symmetric case, $p_i = 1/d$ for $i = 1, \dots, d$.

In Section 3 we give the basic recurrences for the first two moments of the profile and in Section 4 we give our new results. First, we obtain the mean and variance of the external and internal profile. Second, we show the probability function of the depth and height. The results of this paper are derived by methods of analytic combinatorics such as generating functions, Poisson variance and Poissonization.

3 THE RECURRENCES FOR THE FIRST TWO MOMENTS OF PROFILES

The probability generating function of $B_{n,k}$, $P_{n,k}(u) = \mathrm{E}(u^{B_{n,k}})$, satisfies the following recurrence relation

$$P_{n,k}(u) = \sum_{\forall i,k_i \geq 0} \binom{n}{k_1,k_2,...,k_{d-1}} p_1^{k_1} ... p_{d-1}^{k_{d-1}} p_d^{\,n-\sum_{i=1}^{d-1} k_i} P_{k_1,k-1}(u) ... P_{k_{d-1},k-1}(u) P_{n-\sum_{i=1}^{d-1} k_i,k-1}(u), n \geq d, k \geq 1$$

(1)

with initial conditions $P_{0,k}(u) = 1\,(k \geq 0)$, $P_{1,0}(u) = u$, $P_{1,k}(u) = 1\,(k \geq 1)$ and $P_{n,0}(u) = 1\,(n \geq 1)$.

Recall that the first digit determines whether the corresponding string is put to the subtrees. Due to the independence assumption the number of strings where the first digits are $1, \ldots, d-1$ follows a multinomial distribution $M(n, p_1, \ldots, p_{d-1})$. The splitting probabilities are thus given by

$$\binom{n}{k_1, k_2, \ldots, k_{d-1}} p_1^{k_1} \ldots p_{d-1}^{k_{d-1}} p_d^{\,n-\sum_{i=1}^{d-1} k_i}. \tag{2}$$

From (1) one gets directly a recurrence relation for the exponential generating functions

$$M_k(x,u) = \sum_{n \geq 0} P_{n,k}(u) \frac{x^n}{n!} \tag{3}$$

of the form

$$M_k(x,u) = \prod_{i=1}^{d} M_{k-1}(p_i x, u) + (P_{1,k}(u) - P_{1,k-1}(u))x, \quad k \geq 1, \tag{4}$$

with initial condition $M_0(x,u) = e^x + x(u-1)$. With the help of the initial conditions of $P_{1,k}(u)$ it follows that

$$M_1(x,u) = e^x + \sum_{i \neq j}^{d} e^{p_i x} p_j x(u-1) + \sum_{i \neq j}^{d} p_i p_j x^2 (u-1)^2 + (1-u)x \tag{5}$$

and

$$M_k(x,u) = \prod_{i=1}^{d} M_{k-1}(p_i x, u), \qquad k \geq 2. \tag{6}$$

The corresponding probability generating functions for the internal profile, $\overline{P}_{n,k}(u) = \mathrm{E}(u^{I_{n,k}})$ satisfy the same recurrence (1) but with initial conditions $\overline{P}_{n,0}(u) = u\,(n \geq 2)$, $\overline{P}_{n,k}(u) = 1$ for $n \leq 1$ and $k \geq 0$. Similarly to the external profile the exponential generating function

$$\overline{M}_k(x,u) = \sum_{n \geq 0} \overline{P}_{n,k}(u) \frac{x^n}{n!} \tag{7}$$

satisfy

$$\overline{M}_k(x,u) = \prod_{i=1}^{d} \overline{M}_{k-1}(p_i x, u), \qquad k \geq 1, \tag{8}$$

with initial condition $\overline{M}_0(x,u) = ue^x - (1+x)(u-1)$.

Let $\mu_1 = \mathrm{E}[B_{n,k}]$ be the average external profile in d-ary tries and $\mu_2 = \mathrm{E}[B_{n,k}^2]$ its the second moment. By taking the derivative of order 1 with respect to u and setting $u = 1$ from (6) we obtain for the exponential generating function

$$B_k(x) = \sum_{n \geq 0} \mu_1 \frac{x^n}{n!} \tag{9}$$

the following functional recurrence

$$B_k(x) = \sum_{i=1}^{d} e^{p_1 + \ldots + p_{i-1} + p_{i+1} + \ldots + p_d} B_{k-1}(p_i x),\ p_0 := 0. \tag{10}$$

The Poisson transform of $P_k(x) = e^{-x} B_k(x)$ translates recurrence (10) into

$$P_k(x) = \sum_{i=1}^{d} P_{k-1}(p_i x), \qquad k \geq 2, \tag{11}$$

with initial conditions $P_0(x) = xe^{-x}$ and $P_1(x) = \sum_{i \neq j}^{d} e^{-(1-p_i)x} p_j x - xe^{-x}$. By iterating

$$P_k(x)=\sum_{\forall i,k_i\geq 0}^{k-1}\binom{k-1}{k_1,k_2,\ldots,k_{d-1}}P_1\left(p_1^{k_1}\ldots p_{d-1}^{k_{d-1}}p_d^{\,k-1-\sum_{i=1}^{d-1}k_i}x\right),\qquad k\geq 2, \tag{12}$$

where $k_1+\cdots+k_{d-1}=k-1.$ Also

$$\overline{P}_k(x)=\sum_{\forall i,k_i\geq 0}^{k}\binom{k}{k_1,k_2,\ldots,k_{d-1}}\overline{P}_0\left(p_1^{k_1}\ldots p_{d-1}^{k_{d-1}}p_d^{\,k-\sum_{i=1}^{d-1}k_i}x\right),\qquad k\geq 1, \tag{13}$$

where $\overline{P}_0(x)=1-(1+x)e^{-x}$.

By taking the derivative of order 2 with respect to u and setting $u=1$ from (6) we obtain for the exponential generating function

$$D_k(x)=\sum_{n\geq 0}\mu_2\frac{x^n}{n!} \tag{14}$$

the following functional recurrence

$$D_k(x)=dD_{k-1}\left(\frac{x}{d}\right)e^{\frac{x}{d}(d-1)}+d(d-1)\left(B_{k-1}\left(\frac{x}{d}\right)\right)^2e^{\frac{x}{d}(d-2)} \tag{15}$$

and

$$Q_k(x)=e^{-x}D_k(x)=dQ_{k-1}\left(\frac{x}{d}\right)+d(d-1)\left(P_{k-1}\left(\frac{x}{d}\right)\right)^2. \tag{16}$$

Also from (12)

$$P_k(x)=dP_{k-1}\left(\frac{x}{d}\right). \tag{17}$$

Now we apply $V_k(x)=Q_k(x)-P_k^2(x)$ as Poisson variance of the external profile [20]. Thus from (16) and (17),

$$V_k(x)=Q_k(x)-P_k^2(x)=d\left(Q_{k-1}\left(\frac{x}{d}\right)-P_{k-1}^2\left(\frac{x}{d}\right)\right)=dV_{k-1}\left(\frac{x}{d}\right),k\geq 2 \tag{18}$$

and

$$V_1(x)=x^2\left(2\frac{d-1}{d}e^{-x}-\left((d-1)e^{\frac{1-d}{d}x}-e^{-x}\right)^2\right). \tag{19}$$

Let $1\leq s\leq d-1$ and

$$\alpha=\left(-\log\frac{s}{d}\right)^{-1},\beta=\frac{\frac{1}{d}}{-\frac{s^2}{d^2}\log\frac{s}{d}-\frac{(d-s)^2}{d^2}\log\frac{d-s}{d}}. \tag{20}$$

Thus for range $\alpha+\varepsilon\leq\frac{k}{\log n}\leq\beta-\varepsilon$ for some $\varepsilon>0$ with the same consideration of ([12] Theorem 1),

$\mu_1=\mathrm{E}(B_{n,k})\approx Var\ (B_{n,k})\approx V_k(n).$ Also for internal nodes $\overline{\mu}_1=\mathrm{E}(I_{n,k})\approx Var\ (I_{n,k})\approx\overline{V}_k(n).$

4 THE MAIN RESULTS

4.1 Mean and variance of profiles

Lemma 1. For $k\geq 2$,

$$\mathrm{E}(B_{n,k})=n\left((d-1)\left(1+d^{-k}(1-d)\right)^{n-1}-\left(1-d^{1-k}\right)^{n-1}\right) \tag{21}$$

Proof. From (12) for $p_i=\frac{1}{d}$,

$$P_k(x) = x\left((d-1)e^{\frac{(1-d)x}{d^k}} - e^{\frac{-x}{d^{k-1}}}\right), k \geq 2. \tag{22}$$

Let $[x^n]f(x)$ denote the operation of extracting the coefficient of x^n in the formal power series $f(x) = \sum f_n x^n$. We have $B_k(x) = e^x P_k(x).$ Then [9]

$$\mu_1 = \mathrm{E}(B_{n,k}) = n![x^n]B_k(x) = n![x^n]x\left((d-1)e^{\frac{(1-d)x}{d^k}+x} - e^{\frac{-x}{d^{k-1}}+x}\right) \tag{23}$$

and proof is completed.

Lemma 2. For $k \geq 1$,

$$\mathrm{E}(I_{n,k}) = d^k\left(1-\left(1-d^{-k}\right)^n\right) - n\left(1-d^{-k}\right)^{n-1}. \tag{24}$$

Proof. From (13) for $p_i = \frac{1}{d}$,

$$\overline{P}_k(x) = d^k\left(1-\left(1+\frac{x}{d^k}\right)e^{-\frac{x}{d^k}}\right), \quad k \geq 1. \tag{25}$$

Proof is completed just similar to Lemma 1.

If $p_i = 1/d$ for $i = 1, \ldots, d$, then

$$M_k(x,u) = \left(M_{k-1}\left(\frac{x}{d},u\right)\right)^d. \tag{26}$$

Iterating this above equation leads to

$$M_k(x,u) = \left(M_{k-1}\left(\frac{x}{d},u\right)\right)^d = \left(M_1\left(\frac{x}{d^{k-1}},u\right)\right)^{d^{k-1}} = \left((d-1)e^{(1-d)\frac{x}{d^k}}\frac{x}{d^{k-1}} - \frac{x}{d^{k-1}}e^{-\frac{x}{d^{k-1}}}\right)^{d^{k-1}}. \tag{27}$$

Also, $\overline{M}_k(x,u) = \left(\overline{M}_{k-1}\left(\frac{x}{d},u\right)\right)^d$ and iterating

$$\overline{M}_k(x,u) = \left(\overline{M}_{k-1}\left(\frac{x}{d},u\right)\right)^d = \left(\overline{M}_0\left(\frac{x}{d^k},u\right)\right)^{d^k} = \left(ue^{\frac{x}{d^k}} - \left(1+\frac{x}{d^k}\right)(u-1)\right)^{d^k}. \tag{28}$$

Theorem 1. The expected values $\mathrm{E}(B_{n,k})$ and $\mathrm{E}(I_{n,k})$ of the external and internal profile of symmetric d-ary tries are asymptotically given by

$$\mathrm{E}(B_{n,k}) = \begin{cases} n(d-1)\left(1+\frac{1-d}{d^k}\right)^{n-1}, & d^{-k}n \to \infty \\ n\left((d-1)e^{\frac{(1-d)n}{d^k}} - e^{-\frac{n}{d^{k-1}}}\right), & d^{-2k}n \to 0 \end{cases} \tag{29}$$

And

$$\mathrm{E}(I_{n,k}) = \begin{cases} d^k - n\left(1-d^{-k}\right)^{n-1}, & d^{-k}n \to \infty \\ d^k\left(1-\left(1+\frac{n}{d^k}\right)e^{-\frac{n}{d^k}}\right), & d^{-2k}n \to 0. \end{cases} \tag{30}$$

Proof. Proof is completed by de-Poissonization procedures [20].

4.2 Depth and height

The distance from the root to a randomly selected node; its distribution is given by the expected external profile divided by n. That is the distribution of the depth D_n is given by $P(D_n = k) = \frac{\mu_1}{n}$ [4]. Thus from Lemma 1,

$$P(D_n = k) = (d-1)\left(1 + d^{-k}(1-d)\right)^{n-1} - \left(1 - d^{1-k}\right)^{n-1}. \tag{31}$$

The height H_n of tries (the length of the longest path from the root) was already studied by Flajolet and Steyaert [8], Devroye [3], and Pittel [18], and Szpankowski [19]. The limiting behavior was finally determined by Pittel [17]. Let

$$H_k(x) = \sum_{n\geq 0} P(H_n \leq k)\frac{x^n}{n!}. \tag{32}$$

We have (as for the profile)

$$H_k(x) = \prod_{i=1}^{d} H_{k-1}(p_i x), \qquad k \geq 2. \tag{33}$$

It is obvious that $H_1(x) = 1 + x,$ thus

$$H_k(x) = \left(H_{k-1}\left(\frac{x}{d}\right)\right)^d = \left(1 + \frac{x}{d^k}\right)^{d^k}. \tag{34}$$

Hence

$$P(H_n \leq k) = n![x^n]H_k(x) = \frac{n!}{d^{nk}}\binom{d^k}{n} \tag{35}$$

and

$$P(H_n = k) = \frac{n!}{d^{nk}}\binom{d^k}{n} - \frac{n!}{d^{n(k-1)}}\binom{d^{k-1}}{n}. \tag{36}$$

5 DISCUSSION AND CONCLUSIONS

These results are derived by methods of analytic algorithmics such as probability generating function, exponential generating functions, Poissonization and de-Poissonization**.** Our results cover the previous results on 2-ary tries [16].

REFERENCES

[1] Briandais, R. de la. 1959, File searching using variable length keys, *Proceedings of the AFIPS Spring Joint Computer Conference. AFIPS Press, Reston, Va*: 295-298.

[2] Choi, J. D. Lee, K. Loginov, A. O'Callahan, R. Sarkar, V. and Sridharan, M, 2002. Efficient and precise datarace detection for multithreaded object-oriented programs, in Proceedings of the 2002 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI): 258–269.

[3] Devroye, L, 1984. A probabilistic analysis of the height of tries and of the complexity of trie sort. *Acta Inform.*, 1(3): 229–237.

[4] Devroye, L, 2005. Universal asymptotics for random tries and PATRICIA trees, *Algorithmica*, 42: 11-29.

[5] Devroye, L. and Hwang, H.-K, 2006. Width and mode of the profile for some random trees of logarithmic height, *Annals of App. Probab.*, 16: 886-918.

[6] Drmota, M, 2009. *Random Trees, An Interplay Between Combinatorics and Probability*, Springer, Wien-New York.

[7] Drmota, M. and Szpankowski, W, 2011. The expected profile of digital search trees, *J. Comb. Theo, Series A*, 118:1939-1965.

[8] Flajolet, P. and Steyaert, J-M, 1982. A branching process arising in dynamic hashing, trie searching and polynomial factorization. In Automata, languages and programming (Aarhus, 1982), *volume 140 of Lecture Notes in Comput. Sci*., pp. 239–251. Springer, Berlin.

[9] Flajolet, P. and Sedgewick, R, 2008. *Analytic Combinatorics*, Cambridge University Press, Cambridge.

[10] Fredkin, E, 1960 .Trie memory, *Communications of the ACM*, 3: 490–499.

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

[12] Kazemi, R. and Vahidi-Asl, M. Q, 2011. The variance of the profile in digital search trees, *Disc. Math. Theor. Comp. Sci*., 13(3): 21-38.

[13] Knuth, D. E, 1998. *The Art of Computer Programming*, Volume III: Sorting and Searching, 2nd edition, Addison Wesley, Reading, MA.

[14] Kirschenhofer, P. Prodinger, H. and Szpankowski, W, 1996. Analysis of a splitting process arising in probabilistic counting and other related algorithms, *Random Structures and Algorithms*, 9: 379–401.

[15] Mahmoud, H, 1992. *Evolution of random search trees*, John Wiley & Sons Inc., New York.

[16] Park, P. Hwang, H. K. Nicodeme, P. and Szpankowski, W, 2009. Profile of tries, *SIAM Journal on Computing*, 38: 1821-1880.

[17] Pittel, B, 1985. Asymptotical growth of a class of random trees. *Ann. Probab.*, 3(2): 414–427.

[18] Pittel, B, 1986. Paths in a random digital tree: limiting distributions. *Adv. in Appl. Probab.*, 8(1):139–155.

[19] Szpankowski, W, 1991. On the height of digital trees and related problems. *Algorithmica*, (2): 256–277.

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

[21] Ziv, J. and Lempel, A, 1978. Compression of Individual Sequences Via Variable-rate Coding, *IEEE Trans. Inform. Theory*, 24(5):530-536.