TRIES

Wojciech Szpankowski*
Department of Computer Science
Purdue University, W. Lafayette, IN
U.S.A.

April 5, 2008

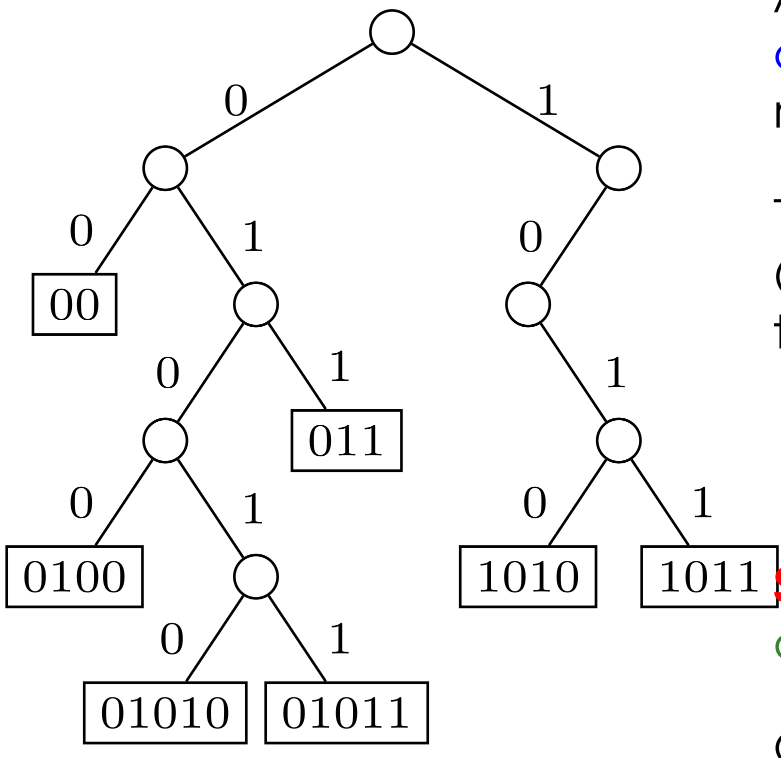
AofA and **IT** logos

*Joint work with L. Devroye, H-K. Hwang, G. Lugosi, P. Nicodeme, G. Park, and M. Ward.

Outline of the Presentation

1. Tries and Suffix Trees
2. Usefulness of Tries
3. Profiles of Tries
   - Trie Parameters
   - Main Results
   - Sketch of Proofs
   - Consequences (height, fill-up, shortest path)
4. Applications: Error Resilient Lempel-Ziv'77 (Suffix Trees)
5. Applications: Distributed Hash Tables (Multiple Tries)

Tries and Suffix Trees



A **trie**, or prefix tree, is an ordered tree data structure that stores keys usually represented by strings.

Tries were introduced by de la Briandais (1959) and Fredkin (1960) who introduced the name:

"tries" derived from **retrieval**.

Suffix tree is a trie built form suffixes of one string.

Other digital trees are: PATRICIA and digital search trees.

Typical Tries: In this talk we mostly discuss random tries built from n (independent) sequences generated by a binary **memoryless source** with p denoting the probability of generating a "0" ($q = 1 - p \leq p$).

Usefulness of Tries

Tries and suffix tress are widely used in diverse applications:

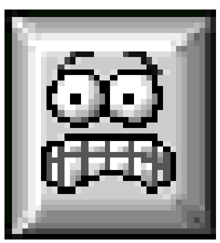
- automatically correcting words in texts; Kukich (1992);
- taxonomies of regular language; Watson (1995);
- event history in datarace detection for multi-threaded object-oriented programs; Choi et al. (2002);
- internet IP addresses lookup; Nilsson and Tikkanen (2002);
- **data compression, Lempel-Ziv**, . . . ; W.S. (2001);
- **distributed hash tables**, Malkhi et al. (2002) and Adler et al. (2003).

Fundamental, prototype data structures:

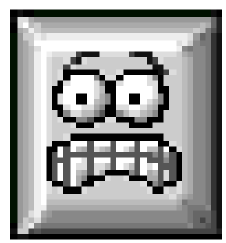
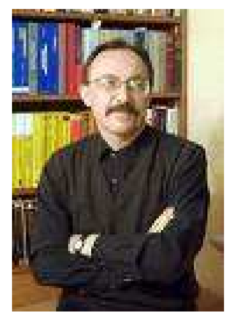
- have a large number of variations and extensions (Patricia, DST, bucket digital search trees, k-d tries, quadtries, LC-tries, multiple-tries, etc.);
- closely connected to several splitting procedures using coin-flipping: collision resolution in multi-access (or broadcast) communication models, loser selection or leader election, etc.
- have direct combinatorial interpretations in terms of words, urn models, etc.

Outline Update

1. Tries and Suffix Trees
2. Usefulness of Tries
3. Profiles of Tries
    - Trie Parameters
    - Main Results
    - Sketch of Proofs
    - Consequences
4. Applications

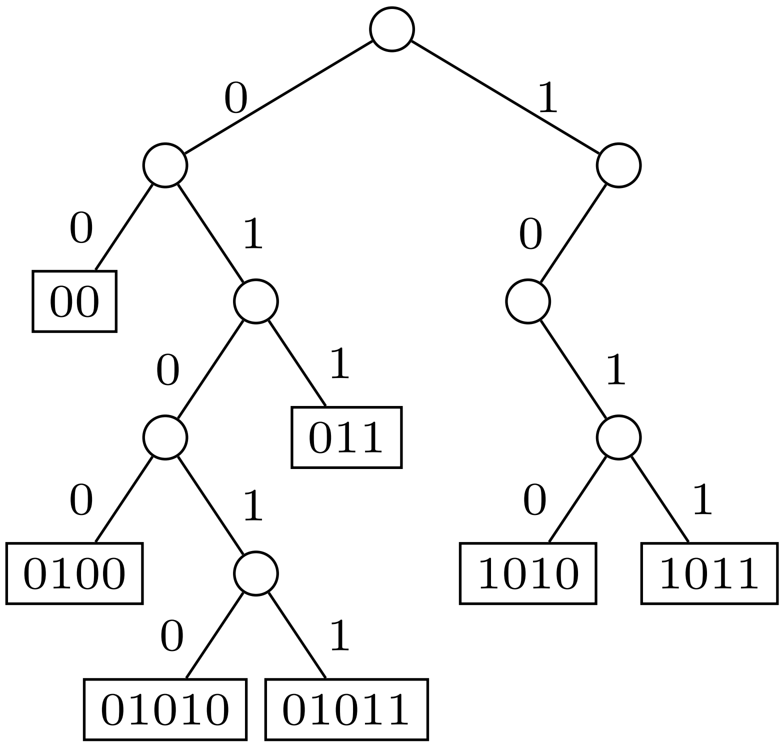



(G. Park)

(P. Nicodeme)

External and Internal Profiles



$B_n^0 = 0, \quad I_n^0 = 1$

$B_n^1 = 0, \quad I_n^1 = 2$

$B_n^2 = 1, \quad I_n^2 = 2$

$B_n^3 = 1, \quad I_n^3 = 2$

$B_n^4 = 3, \quad I_n^4 = 1$

$B_n^5 = 2 \quad I_n^5 = 0$

External profile and internal profile:

$B_n^k = \#$ external nodes at distance k from the root;

$I_n^k = \#$ internal nodes at distance k from the root.

Why to Study Profiles?

- Fine, informative shape characteristic;
- Related to path length, depth, height, shortest path, width, etc.;
- Breadth-first search;
- Compression algorithms.
- Mathematically challenging, phenomenally interesting!

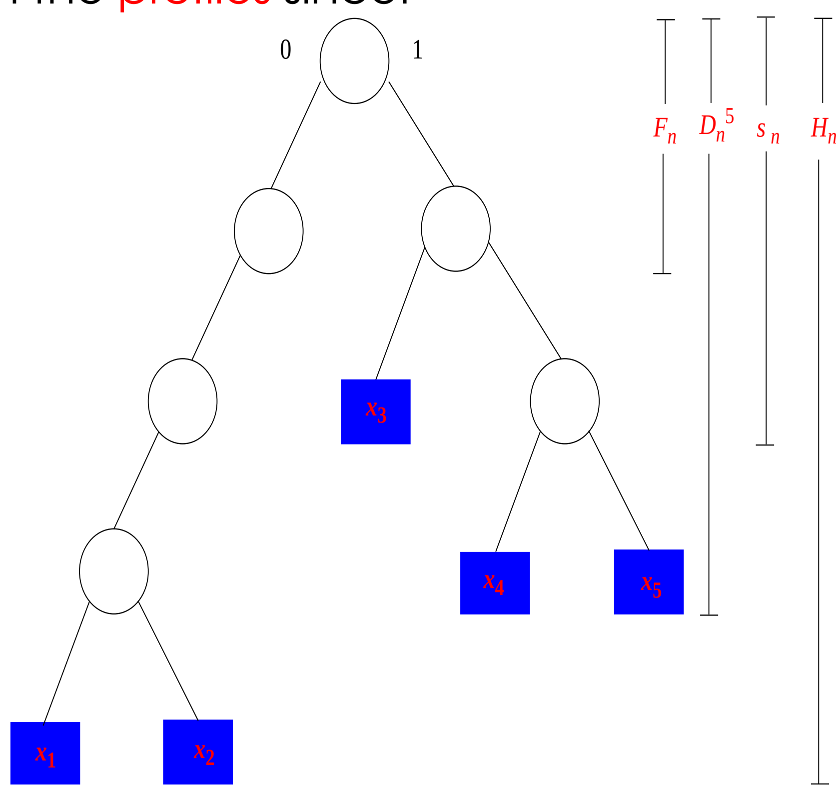
Example: Parameters such height H_n, shortest path, s_n, fill-up level F_n, and depth, D_n can be studied through the profiles since:

$$H_n = \max\{k : B_n^k > 0\},$$

$$s_n = \min\{k : B_n^k > 0\},$$

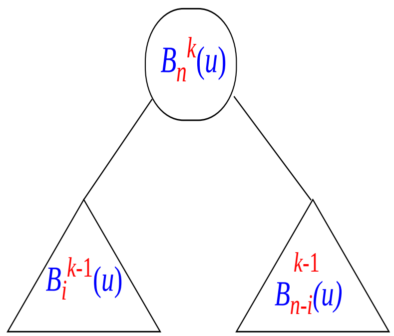
$$F_n = \max\{k : I_n^k = 2^k\},$$

$$\Pr(D_n = k) = \frac{\mathbb{E}[B_n^k]}{n}.$$



Recurrence for the Profiles

External Profile B_n^k:
Define the probability generating function as



$$B_n^k(u) = \mathbb{E}[u^{B_n^k}] = \sum_{\ell \geq 0} P(B_n^k = l)u^l.$$

Then

$$B_n^k(u) = \sum_{i=0}^{n} \binom{n}{i} p^i q^{n-i} B_i^{k-1}(u) B_{n-i}^{k-1}(u)$$

with $B_n^0 = 1$ for $n \neq 1$ and $B_1^0 = u$

Internal Profile probability generating function $I_n^k(u) = \mathbb{E}[I_n^k]$ satisfies the same recurrence with $U_n^0(u) = u$ for $n > 1$ and $U_0^0(u) = U_1^0(u) = 1$.

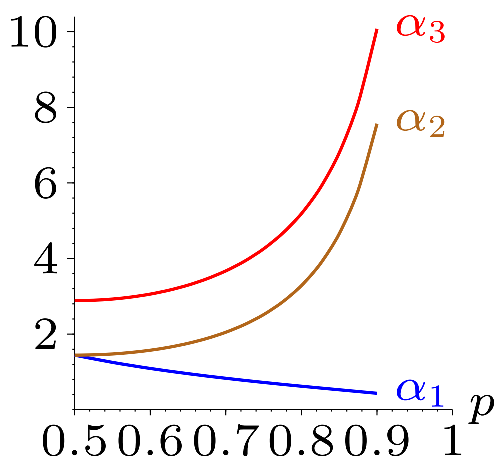
Average External Profile:

$$\boxed{\mathbb{E}[B_n^k] = \sum_{i=0}^{n} \binom{n}{i} p^i q^{n-i} (\mathbb{E}[B_i^{k-1}] + \mathbb{E}[B_{n-i}^{k-1}]), \;\; n \geq 2, k \geq 1,}$$

under some initial conditions (e.g., $\mathbb{E}[B_0^k] = 0$ for all k).

Main Results

Notation: $r = p/q = p/(1-p) > 1$, and $\alpha := \alpha_{n,k} = \frac{k}{\log n}$. Also:



$$\alpha_1 := \frac{1}{\log(1/q)},$$

$$\alpha_2 := \frac{p^2+q^2}{p^2\log(1/p)+q^2\log(1/q)},$$

$$\alpha_3 := \frac{2}{\log(1/(p^2+q^2))}.$$

1: **Exponential Growth** $(0<\alpha<\alpha_1)$:

Let $1 \le k \le \frac{1}{\log q^{-1}}(\log n - \log\log\log n + \log(r-1) - \varepsilon)$:

$$\mathbb{E}[B_n^k] = nq^k(1-q^k)^{n-1}\left(1+O\left((\log n)^{-\delta}\right)\right) = O(2^{-n\nu})$$

2: **Logarithmic Growth** $(0<\alpha<\alpha_1)$:

Let $1 \le k \le \frac{1}{\log q^{-1}}(\log n - \log\log\log n + m\log(r-1) - \varepsilon)$:

$$\mathbb{E}[B_n^k] = O(\log\log n \cdot \log^{m-\beta} n).$$

where m and β are constants (smaller or greater than m).

Phase Transitions

3: **Polynomial Growth**: $\alpha_1 \cdot \log n < k < \alpha_2 \cdot \log n$: $\quad (\alpha_1 < \alpha < \alpha_2)$

$$\mathbb{E}[B_n^k] \sim G_1\,(\log n) \frac{p^\rho q^\rho (p^{-\rho} + q^{-\rho})}{\sqrt{2\pi\alpha}\ \log(p/q)} \cdot \frac{n^{\upsilon_1}}{\sqrt{\log n}},$$

where $G_1(x)$ is a periodic function and

$$\upsilon_1 = -\rho + \alpha \log(p^{-\rho} + q^{-\rho}), \qquad \rho = -\frac{1}{\log(p/q)} \log\left(\frac{-1 - \alpha \log q}{1 + \alpha \log p}\right).$$

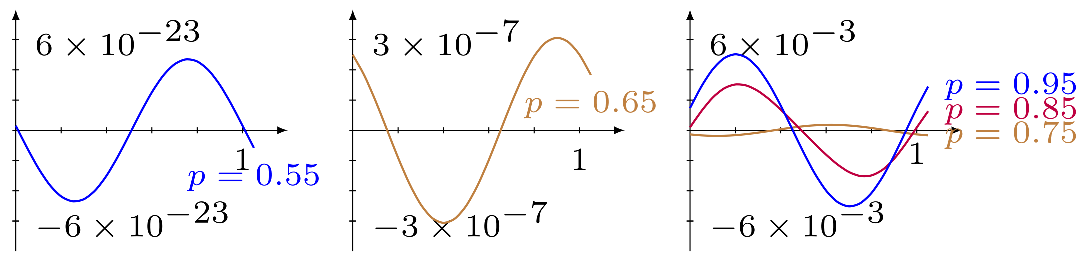


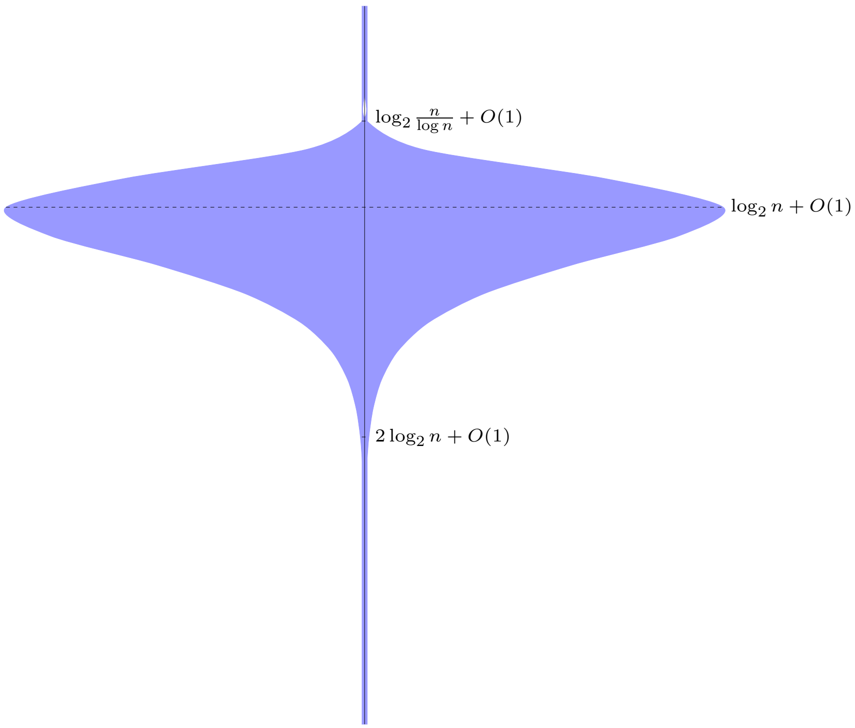
Figure 1: *The fluctuating part of the periodic function $G_1(x)$ for $p = 0.55, 0.65, \ldots, 0.95$.*

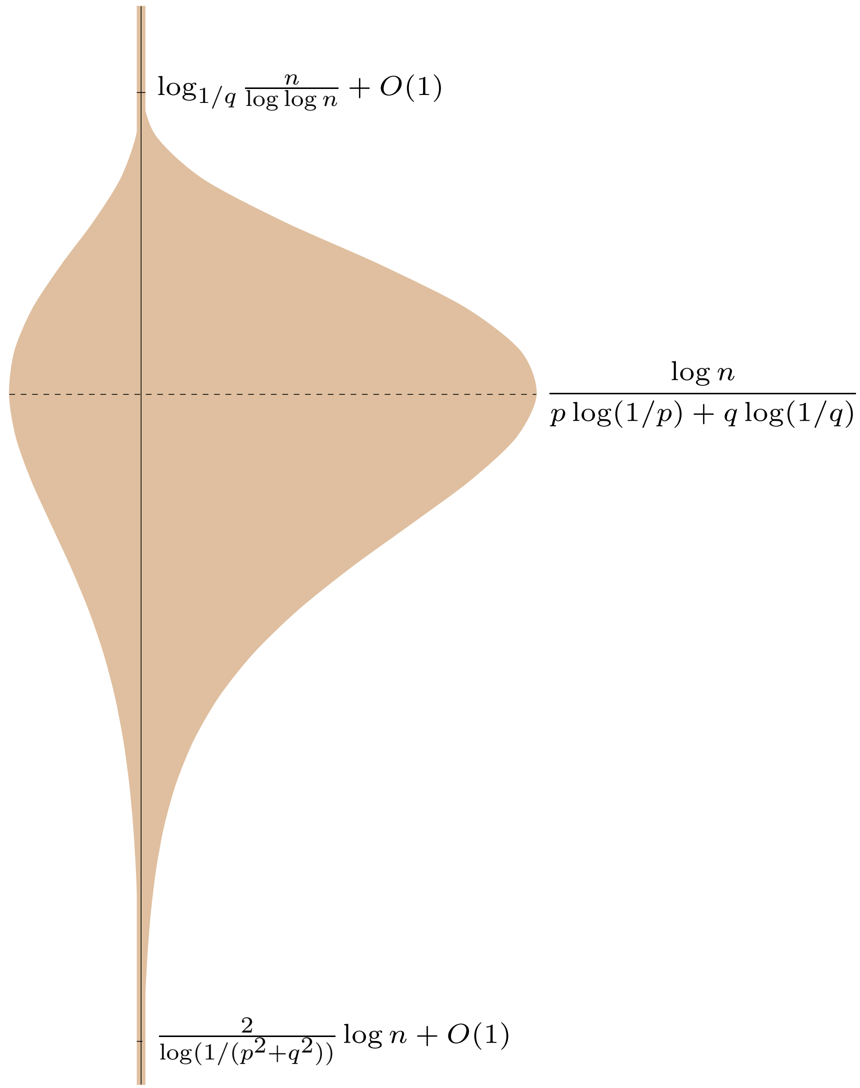
4: **Polynomial Growth/Decay**: $\alpha_2 \cdot \log n < k$: $\quad (\alpha_2 < \alpha)$

$$\mathbb{E}[B_n^k] = \frac{2pq}{p^2 + q^2} n^{\nu_2} + O\left(n^{\nu_3}\right)$$

where $\nu_2 = 2 + \alpha \log(p^2 + q^2)$ for some $\nu_3 < \nu_2$.

External Shapes





$(p = 0.5, \alpha_1 = \alpha_2 = 1/\log 2)$ $(p = 0.75)$

Average Internal Profile

1: **Almost Full Tree**: $k < \alpha_1 \cdot \log n$

$$\mathbb{E}(I_n^k) = 2^k - \mathbb{E}(B_{n,k})(1 + o(1)).$$

2: **Phase Transition I**: $\alpha_1 \cdot \log n < k < \alpha_0 \cdot \log n$, where $\alpha_0 = \frac{2}{\log(1/p)+\log(1/q)}$

$$\mathbb{E}[I_n^k] = 2^k - G_2(\log n)\mathbb{E}(B_{n,k})(1 + o(1))$$

where $G_2(x)$ is a periodic function.

3: **Phase Transition II**: $\alpha_0 \cdot \log n < k < \alpha_2 \cdot \log n$

$$\mathbb{E}[I_n^k] = G_2(\log n)\mathbb{E}(B_{n,k})(1 + o(1))$$

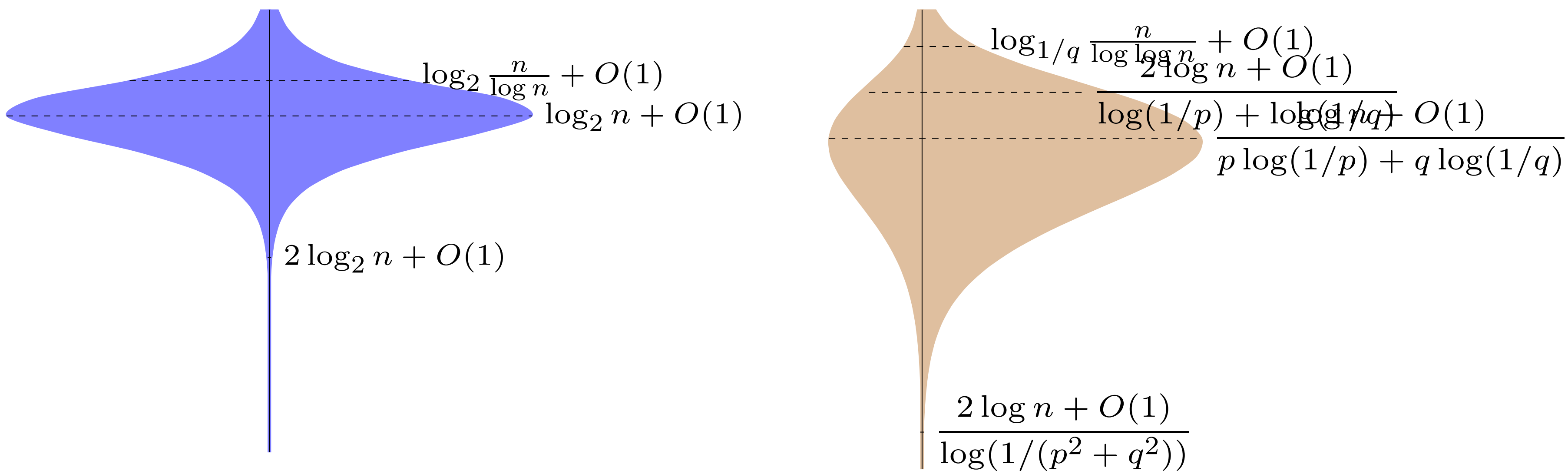
where $G_2(x)$ is a periodic function.

4: **Polynomial Growth/Decay**: $\alpha_2 \cdot \log n < k$

$$\mathbb{E}[I_n^k] = \frac{1}{2} n^{\nu_2}(1 + o(1))$$

where $\nu_2 = 2 - \alpha \log(p^2 + q^2)$.

Internal Shapes



$(p = 0.5\ \alpha_0 = \alpha_1 = \alpha_2 = 1/\log 2)$ $(p = 0.75)$

Variance and Limiting Distributions of the External Profile

Variance:

1: $k < \alpha_1 \cdot \log n$: $\mathbb{V}[B_n^k] \sim \mathbb{E}[B_n^k]$.

2: $\alpha_1 \cdot \log n < k < \alpha_2 \cdot \log n$: $\mathbb{V}[B_n^k] \sim G_3(\log n)\mathbb{E}[B_n^k]$.
where $G_3(\log n)$ is a periodic function.

3: $\alpha_2 \cdot \log n < k$: $\mathbb{V}[B_n^k] \sim 2\mathbb{E}[B_n^k]$.

Limiting Distributions:
Central Limit Theorem: For $\alpha_1 \cdot \log n < k < \alpha_3 \cdot \log n$:

$$\frac{B_n^k - \mathbb{E}[B_n^k]}{\sqrt{\mathbb{V}[B_n^k]}} \to N(0,1),$$

where $N(0,1)$ is the standard normal distribution.

Poisson Distribution: For $\alpha_3 \cdot \log n < k$:

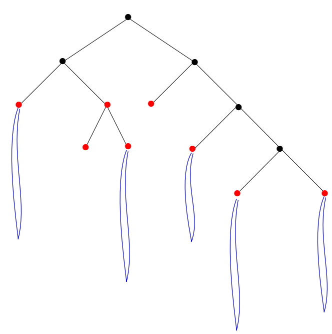
$$P\left(B_{n,k} = 2m\right) = \frac{{\lambda_0}^m}{m!} e^{-\lambda_0} + o(1), \quad \text{and} \quad P\left(B_{n,k} = 2m+1\right) = o(1),$$

where $\lambda_0 := pqn^2(p^2 + q^2)^{k-1}$.

Consequences

Height: For large n (cf. Flajolet, 1980, Pittel, 1985, W.S., 1988, Devroye, 1992)

$$H_n = \frac{2}{\log(p^2+q^2)^{-1}}\log n = \alpha_3 \log n := k_H, \quad (\text{whp}).$$



Upper Bound: $P(H_n > (1+\epsilon)k_H) \leq P(B_n^k \geq 1) \leq \mathbb{E}[B_n^k] \to 0.$

Lower Bound: $P(H_n < (1-\epsilon)k_H) \leq P(B_n^{\lceil(1-\varepsilon)k_H\rceil}) = 0)$

$$\leq \frac{\mathbb{V}[B_n^{\lceil(1-\epsilon)k_H\rceil}]}{(\mathbb{E}[B_n^{\lceil(1-\epsilon)k_H\rceil}])^2} = O\left(\frac{1}{\mathbb{E}[B_n^{\lceil(1-\epsilon)k_H\rceil}]}\right) \to 0.$$

Define: $k_S := \lfloor \frac{1}{\log q^{-1}}(\log n - \log\log\log n + \log(e \log r))\rfloor.$

Shortest Path: For large n (cf. Knessl and W.S., 2005)

$$P(s_n = k_S \text{ or } s_n = k_S + 1) \to 1.$$

Fill-up: For large n (cf. Pittel, 1986, Devroye, 1992, Knessl & W.S., 2005)

$$P(F_n = k_S - 1 \text{ or } F_n = k_S) \to 1.$$

Sketch of the Proof

1. **Recurrence**: $\mathbb{E}[B_n^k] = \sum_{i=0}^n \binom{n}{i} p^i q^{n-i} (\mathbb{E}[B_i^{k-1}] + \mathbb{E}[B_{n-i}^{k-1}]),\, n \geq 2,\, k \geq 1.$

2. **Poisson Transform**: $\tilde{E}_k(z) = \sum_{n=0}^\infty \mathbb{E}[B_n^k] \frac{z^n}{n!} e^{-z}$:

$$\tilde{E}_k(z) = \tilde{E}_{k-1}(zp) + \tilde{E}_{k-1}(zq),\ \ k \geq 2,$$

3. **Mellin Transform**: $\tilde{E}_k^*(s) := \int_0^\infty z^{s-1} \tilde{E}_k(z) dz = (p^{-s} + q^{-s}) \tilde{E}_{k-1}^*(s)$:

$$\tilde{E}_k^*(s) = (p^{-s} + q^{-s})^{k-1} \cdot s \cdot (p^{-s} + q^{-s} - 1)\Gamma(s)$$

for $\Re(s) \in (-2, \infty)$, where $\Gamma(s)$ is the Euler Gamma function.

4. **Inverse Mellin Transform**: $\tilde{E}_k(z) = \frac{1}{2\pi i} \int_{c-i\infty}^{c+i\infty} z^{-s} \tilde{E}_k^*(s) ds$:

$$\tilde{E}_k(z) = \frac{1}{2\pi i} \int_{c-i\infty}^{c+i\infty} s(p^{-s} + q^{-s} - 1)\Gamma(s) z^{-s} (p^{-s} + q^{-s})^{k-1} ds$$

through the **saddle point method**.

5. **Depoissonization**: From the Poisson transform $\tilde{E}_k(z)$ to $\mathbb{E}[B_n^k]$.

Saddle Point Method: Phase Transitions

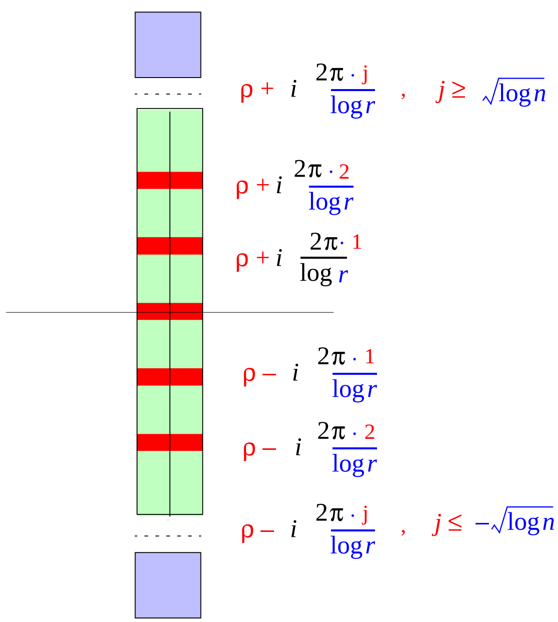
By depoisonization we have $\tilde{E}_k(n) \sim \tilde{E}_k(z)$, where recall

$$\tilde{E}_k(n) = \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} g(s)\Gamma(s+1)n^{-s}(p^{-s}+q^{-s})^k ds$$
$$= \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} g(s)\Gamma(s+1)\exp(h(s)\log n)ds, \qquad k=\alpha\log n.$$

The saddle point equation $h'(s)=0$ has a unique real root:

$$\rho = \frac{-1}{\log r}\log\left(\frac{\alpha\log q^{-1}-1}{1-\alpha\log p^{-1}}\right), \qquad \frac{1}{\log q^{-1}} < \alpha < \frac{1}{\log p^{-1}}.$$

There are **infinitely many** saddle points $\rho + it_j$ for $t_j = 2\pi j/\log r$, $j\in\mathbb{Z}$.



Phase Transitions:

1. $\rho\to\infty$ as $\alpha\downarrow 1/\log q^{-1} = \alpha_1$.

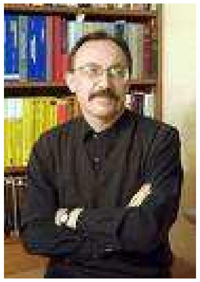
2 $\rho\to-\infty$ when $\alpha\uparrow 1/\log p^{-1}$.

3. **Saddle points coalesce** with poles of the $\Gamma(s+1)$ function at $s=-2,-3,\ldots$.
Pole $s=-2$ leads to α_2.

Outline Update

1. Tries and Suffix Trees
2. Usefulness of Tries
3. Profiles of Tries
4. Applications: Error Resilient Lempel-Ziv'77 (Suffix Trees)
5. Application: Distributed Hash Tables (Multiple Tries)





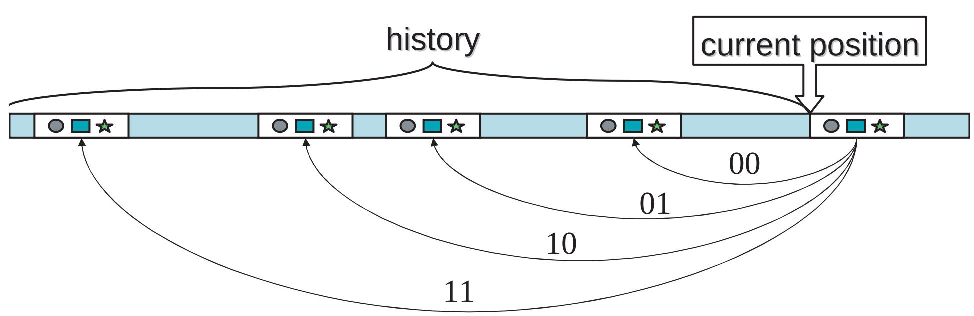
Error Resilient LZ'77 Scheme

1. The **Lempel-Ziv'77** works on-line: It compresses phrases by replacing the longest prefix by (pointer, length) of its copy.

2. Castelli and Lastras in 2004 proved that a single error in LZ'77 corrupts $O(n^{2/3})$ phrases, thus about $O(n^{2/3}\log n)$ symbols, where n is the size.

3. There are **multiple copies** of the longest prefix that we denote by M_n for a database of length n.

4. By a judicious choice of pointers in the LZ'77 scheme, we can recover $\lfloor \log_2 M_n \rfloor$ bits **without losing a bit in compression**. Parity bits recovered from the multiple copies are used for the Reed-Solomon channel coding.



Experimental Results: I

Table 1: The compression of "`gzip -3`" (we also call it LZS'77) versus "`gzipS -3`" for the files of the Calgary corpus; the last column shows the total number of available bytes for error correction.

file size	gzip	gzipS	*file*	*redundant*
111,261	39,473	39,511	bib	1,721
768,771	333,776	336,256	book1	14,524
610,856	228,321	228,242	book2	10,361
102,400	69,478	71,168	geo	4,101
377,109	155,290	156,150	news	5,956
21,504	10,584	10,783	obj1	353
246,814	89,467	89,757	obj2	3,628
53,161	20,110	20,204	paper1	937
82,199	32,529	32,507	paper2	1,551
46,526	19,450	19,567	paper3	893
13,286	5,853	5,898	paper4	249
11,954	5,252	5,294	paper5	210
38,105	14,433	14,506	paper6	738
513,216	62,357	61,259	pic	3,025
39,611	14,510	14,660	progc	736
71,646	18,310	18,407	progl	1,106
49,379	12,532	12,572	progp	741
93,695	22,178	22,098	trans	1,201

Encoder and Decoder of LZRS'77

We use the family of Reed-Solomon codes RS$(255, 255 - 2e)$ that contains blocks of 255 bytes, of which $255 - 2e$ are data and $2e$ are parity.

Encoder: The data is broken into blocks of size $255 - 2e$. Blocks are processed in reverse order, beginning with the very last. When processing block i, the encoder computes first the Reed-Solomon parity bits for the block $i + 1$ and then it embeds the extra bits in the pointers of block i.

Decoder: The decoder receives a sequence of pointers, preceded by the parity bits of the first block which are used to correct block B_1. Once block B_1 is correct, it decompresses it using LZS'77. Redundant bits of block B_1 are used as parity bits to correct block B_2, etc.

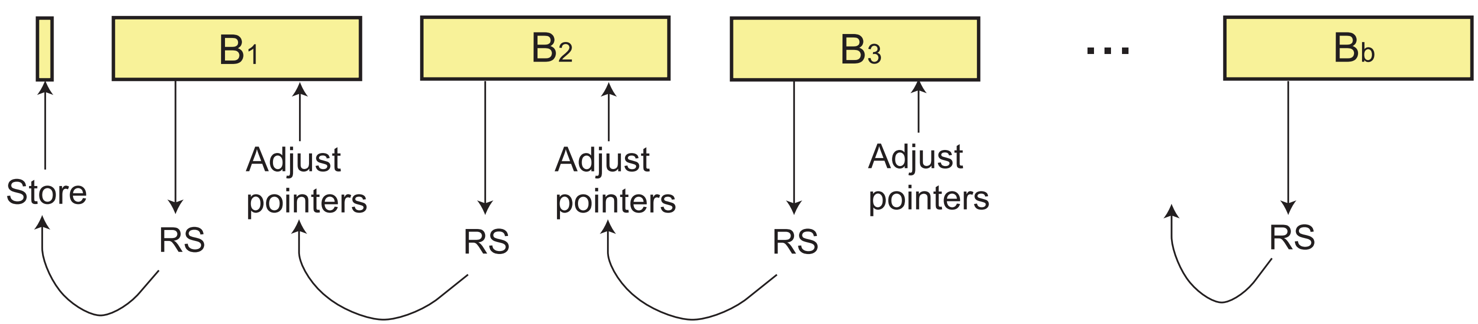


Figure 2: The right-to-left sequence of operations on the blocks.

Analysis of M_n Via Suffix Trees

Why does LZRS'77 work so well?
Performance of LZRS'77 depends on M_n. How does M_n typically behave?
Build a **suffix tree** from the first n suffixes of the database X (i.e., $S_1 = X_1^\infty, S_2 = X_2^\infty, \ldots, S_n = X_n^\infty$). Then insert the $(n+1)$st suffix, $S_{n+1} = X_{n+1}^\infty$.

Depth of insertion of S_{n+1} is the $(n+1)$-st phrase length, and M_n is the size of the subtree that starts at the insertion point of the $(n+1)$st suffix.

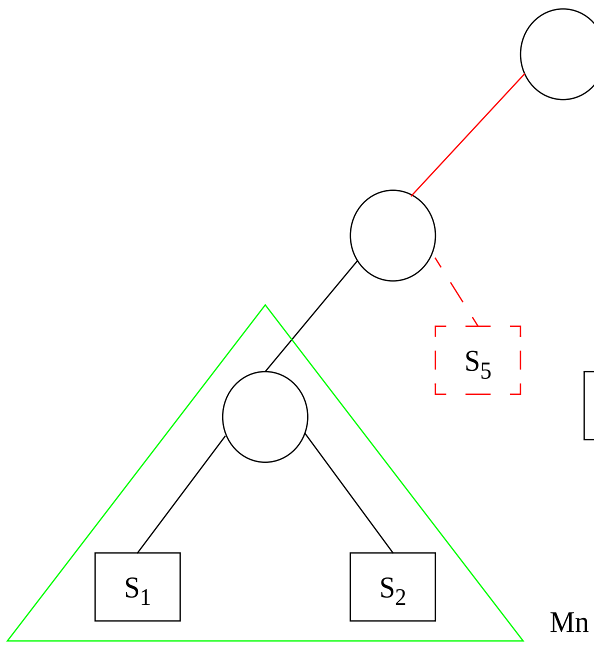
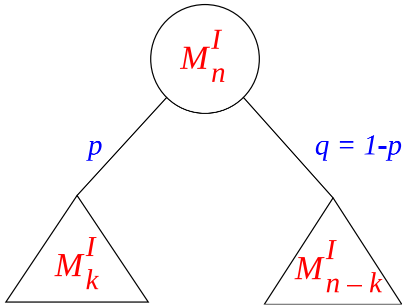


Figure 4: M_4(=2) is the size of the subtree at the insertion point of S_5.

Analysis of M_n for Independent Tries

1. Consider **digital tries** built over n **independent** strings.
Average $\mathbb{E}[M_n^I]$ and probability generating function
satisfying the following recurrences
($p = 1 - q$ is the probability of generating a "1")



$$\mathbb{E}[M_n^I] = p^n(qn + p\mathbb{E}[M_n^I]) + q^n(pn + q\mathbb{E}[M_n^I)] + \sum_{k=1}^{n-1} \binom{n}{k} p^k q^{n-k} (p\mathbb{E}[M_k^I] + q\mathbb{E}[M_{n-k}^I])$$

$$\mathbb{E}[u^{M_n^I}] = p^n(qu^n + p\mathbb{E}[u^{M_n^I}]) + q^n(pu^n + q\mathbb{E}[u^{M_n^I}]) + \sum_{k=1}^{n-1} \binom{n}{k} p^k q^{n-k} (p\mathbb{E}[u^{M_k^I}] + q\mathbb{E}[u^{M_{n-k}^I}]).$$

- **(Analytic) Poissonization** $\left(\widetilde{W}(z) = \sum_{n\geq 0} \mathbb{E}[M_n^I]\frac{z^n}{n!}e^{-z}\right)$:

$$\widetilde{W}(z) = qpze^{qz} + pqze^{pz} + p\widetilde{W}(pz) + q\widetilde{W}(qz).$$

- **Mellin Transform** $\left(f^*(s) = \int_0^\infty f(x)x^{s-1}dx\right)$:

$$W^*(s) = \frac{\Gamma(s+1)(pq^{-s} + qp^{-s})}{1 - p^{-s+1} - q^{-s+1}}.$$

- **Inverse Mellin Transform**: $\widetilde{W}(z) = 1/h + \text{fluctuations}.$
- **(Analytic) Depoissonization**: $\mathbb{E}[M_n^I] = 1/h + \text{fluctuations}.$

Analysis of M_n for Dependent Strings

2. Suffix Trees: Using **analytic combinatorics on words** we prove that

$$M(z,u) = \sum_{n=1}^{\infty}\sum_{k=1}^{\infty} \mathbf{P}(M_n = k) u^k z^n$$
$$= \sum_{\substack{w\in\mathcal{A}^* \\ \alpha\in\mathcal{A}}} \frac{u\mathbf{P}(\beta)\mathbf{P}(w)}{D_w(z)} \frac{D_{w\alpha}(z) - (1-z)}{D_w(z) - u(D_{w\alpha}(z) - (1-z))}$$

$D_w(z) = (1-z)S_w(z) + z^m P(w)$ and $S_w(z)$ is the autocorrelation polynomial:

$$S_w(z) = \sum_{k\in\mathcal{P}(w)} \mathbf{P}(w_{k+1}^m) z^{m-k}$$

$\mathcal{P}(w)$ denotes the set of positions k of w satisfying $w_1 \ldots w_k = w_{m-k+1} \ldots w_m$.

For any $\varepsilon > 0$ there exists $\beta > 1$ such that (all hard analytic work is here!)

$$|\Pr(M_n = k) - \Pr(M_n^I = k)| = O(n^{-\varepsilon}\beta^{-k})$$

Random suffix trees resemble random independent tries (cf. P. Jacquet, W.S., 1994, Lonardi, W.S., Ward, 2005: IEEE Trans. Inf. Th., 2007).

Main Results

Theorem 2 (Ward, W.S., 2005). *Let* $z_k = \frac{2kr\pi i}{\ln p}$ $\forall k \in \mathbb{Z}$, *where* $\frac{\ln p}{\ln q} = \frac{r}{s}$ *for some relatively prime integers* r, s *(i.e.,* $\frac{\ln p}{\ln q}$ *is rational).*
The j*th factorial moment* $\mathbb{E}[(M_n)^{\underline{j}}] = \mathbb{E}[M_n(M_n-1)\cdots M_n(-j+1)]$ *is*

$$\mathbb{E}[(M_n)^{\underline{j}}] \quad = \quad \Gamma(j)\frac{q(p/q)^j + p(q/p)^j}{h} + \delta_j(\log_{1/p} n) + O(n^{-\eta})$$

where $h = -p\log p - q\log q$ *is the entropy rate,* $\eta > 0$*, and where* Γ *is the Euler gamma function and*

$$\delta_j(t) = \sum_{k\neq 0} -\frac{e^{2kr\pi it}\Gamma(z_k+j)\left(p^j q^{-z_k-j+1} + q^j p^{-z_k-j+1}\right)}{p^{-z_k+1}\ln p + q^{-z_k+1}\ln q}.$$

δ_j *is a periodic function that has a small magnitude and exhibits fluctuation when* $\frac{\ln p}{\ln q}$ *is rational*

Note: On average there are $\mathbb{E}[M_n] \sim 1/h$ additional pointers.

j	$\frac{1}{\ln 2}\sum_{k\neq 0}\left\|\Gamma\left(j - \frac{2ki\pi}{\ln 2}\right)\right\|$
1	1.4260 $\times 10^{-5}$
3	1.2072 $\times 10^{-3}$
5	1.1421 $\times 10^{-1}$
6	1.1823 $\times 10^{0}$
8	1.4721 $\times 10^{2}$
9	1.7798 $\times 10^{3}$
10	2.2737 $\times 10^{4}$

Distribution of M_n

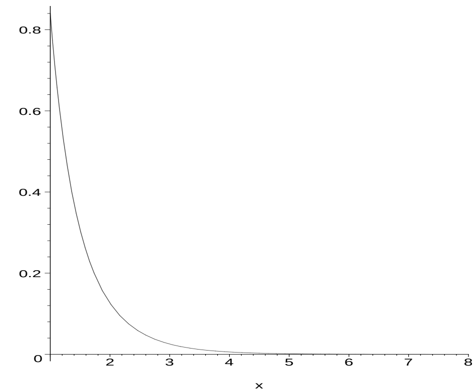
Theorem 3 (Ward, W.S., 2005). *Let* $z_k = \frac{2kr\pi i}{\ln p}$ $\forall k \in \mathbb{Z}$, *where* $\frac{\ln p}{\ln q} = \frac{r}{s}$. *Then*

$$P(M_n = j) \quad = \quad \frac{p^j q + q^j p}{jh} + \sum_{k \neq 0} - \frac{e^{2kr\pi i \log_{1/p} n} \Gamma(z_k)(p^j q + q^j p)(z_k)^{\overline{j}}}{j!(p^{-z_k+1} \ln p + q^{-z_k+1} \ln q)} + O(n^{-\eta})$$

where $\eta > 0$, *and* Γ *is the Euler gamma function.*

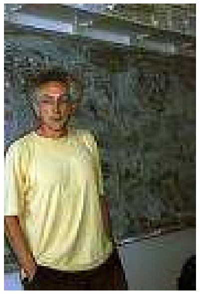
Therefore, M_n *follows the logarithmic series distribution with mean* $1/h$ *(plus some fluctuations).*

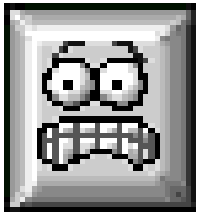
The logarithmic series distribution ($(p^j q + q^j p)/(jh)$)
is well concentrated around its mean $\mathbb{E}M_n \approx 1/h$.



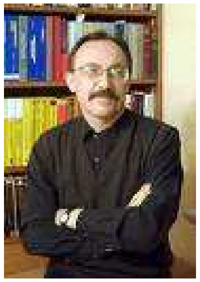
Outline Update

1. Tries and Suffix Trees
2. Why to bother?
3. Profiles of Tries
4. Applications: Error Resilient Lempel-Ziv'77
5. Application: Distributed Hash Tables (Multiple Tries)





(G. Park)



Distributed Hashing Tables

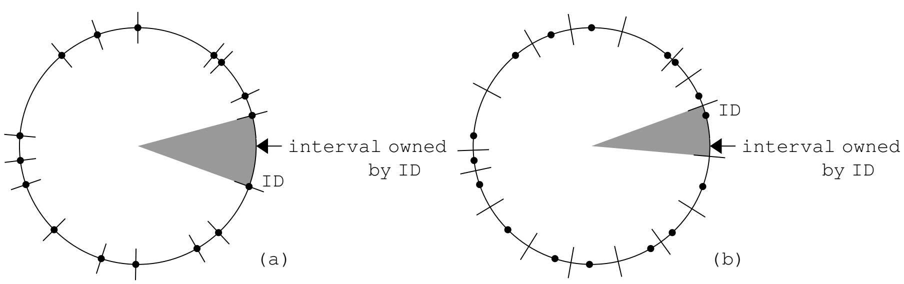
1. Distributed Hash Table (DHT) is a decentralized distributed system that provides a lookup service similar to hash table by maintaining the mapping from names to values distributed among the nodes.

2. DHT is used in peer-to-peer networks (e.g., Napster, Gnutella and Freenet) for ID/keys management.

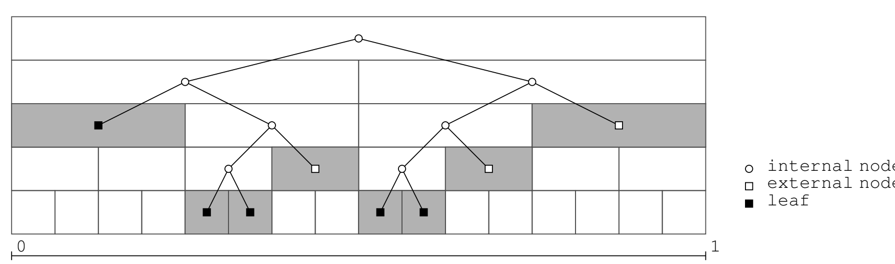
3. Each of n users/processors is given a key that is mapped into the unit interval $[0, 1]$ forming a unit circle.

4. Users can be considered as (infinite) binary strings which often are organized as in a binary trie: strings are i.i.d. with $p = q = 1/2$.

5. The trie is used to locate peers with closest keys. It also partitions the unit circle into intervals, which are owned by a host either to its left or determined by the virtue of the trie.



(a)



(b)

Parameters of DHT

The objective of ID management is to make all intervals of about equal length: Two parameters are important:

(a) Balance B_n of the partition determined by the length of the largest and smallest interval.

(b) Search Time to locate the host that owns an interval.

For a trie implementation of DHT, the balance B_n can be defined as

$$B_n = 2^{H_n - F_n + O(1)}.$$

The goal is to make $B_n = O(1)$.

We saw that for a symmetric trie ($p = q = 1/2$)

$$H_n = 2\log_2 n + O(1), \quad F_n = \log_2 n - \log_2 \log_2 n + O(1).$$

so that

$$B_n = 2^{2\log n - \log 2 + \log\log n + O(1)} = n + O(\log n).$$

Two-choice Tries

Goal: Build a well-balanced trie with height close to its fillup level.

Two-choice Trie:
Each datum (key) has two strings, X_i and Y_i, that is, there are n pairs of strings (X_i, Y_i), and we can select one, say Z_i of the two to insert in the trie.

A Greedy Heuristic:
Choose the string which, at the time of insertion would yield the leaf nearest to the root. (Once the selection is made, it cannot be undone!)

Main Results:
With high probability

$$\frac{H_n}{\log n} \to \frac{3}{2}\frac{1}{Q} = \frac{3}{4}\alpha_2$$

where $Q = -\log P_2 = -\log(p^2 + q^2)$.

Thus the height is reduced by 25% when compared to a standard trie and

$$B_n = 2^{\log\sqrt{n} - O(\log\log n)} = \sqrt{n} + O(\log n).$$

Optimal off-line Algorithm

Define: $Z_i(0) = X_i$ and $Z_i(1) = Y_i$. Let $\{i_1, \ldots, i_n\} \subseteq \{0,1\}^n$ such that

$H_n(i_1, \ldots, i_n)$ is the height over $Z_1(i_1), \ldots, Z_n(i_n)$.

Define the optimal height

$$H_n^* = \min_{i_1,\ldots,i_n} H_n(i_1, \ldots, i_n)$$

over all 2^n tries.

Main Results:
With high probability

$$\frac{H_n}{\log n} \to \frac{1}{Q} = \frac{1}{2}\alpha_2.$$

More precisely:

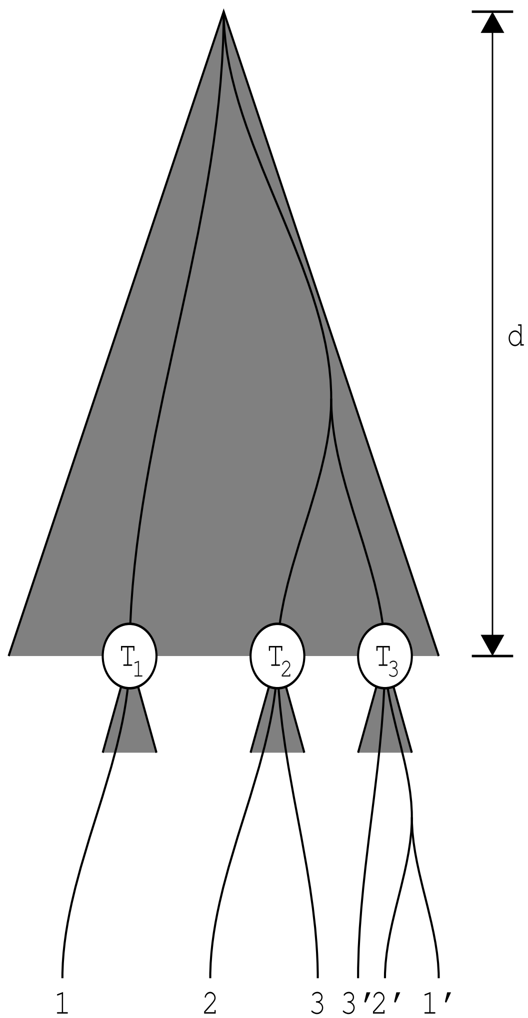
$$\mathbb{P}\left\{H_n^* \geq \frac{\log n \ + t}{Q}\right\} \leq 8e^{-t}, \quad \lim_{n\to\infty} \mathbb{P}\left\{H_n^* \leq \frac{(1-\epsilon)\log n}{Q}\right\} = 0.$$

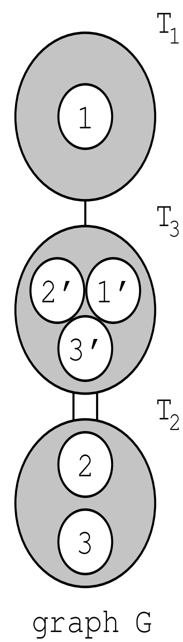
Thus the height is reduced by 50% when compared to standard trie and

$$B_n = 2^{\log\log n + O(1)} = \log n + O(1).$$

Sketch of the Proof: Construction of an Optimal Trie

Construct an infinite trie over $2n$ strings. How to select n strings?





1. Let T_j $(1 \le j \le 2^d)$ be a subtree rooted at distance d from the root.

2. A bad datum is when both strings (of the same datum) fall in the same T_j.

3. A colliding pair of data is such that for $j \neq k$, each datum in the pair delivers one string to T_j and one string to T_k.

4. We construct a multigraph $G(d)$ whose vertices represent the T_j. We connect T_j with T_ℓ if a datum deposits one string in each of these trees.

5. Our job to select n strings in $G(d)$ so that there is no: (i) bad datum, (ii) no colliding data, (iii) and no cycle.

Sketch of the Proof: Algorithm

Algorithm:
1. We choose any one of the strings in the root node's list.
2. For all other strings, choose the companion string of the same datum.
3. Continue until one string of each datum is chosen for the trie.

Lemma 1. (i) *The probability of bad datum anywhere is not more than:*

$$nP_2^{\,d}, \quad P_2 = p^2 = q^2.$$

(ii) *The probability of a colliding pair anywhere is not more than:* $2n^2 P_2^{\,2d}$.
(iii) *The probability that* G *has a cycle of length* ≥ 3 *is not more than:*

$$\frac{(4n)^3 P_2^{\,3d}}{1 - 4nP_2^{\,d}}.$$

Proof of Main Result:

$$\begin{aligned}
\mathbb{P}\{H_n > d\} &\leq \mathbb{P}\{\text{there exists a bad datum}\} + \mathbb{P}\{\text{there exists a colliding pair}\} \\
&\quad + \mathbb{P}\{\text{there exists a cycle}\} \\
&\leq nP_2^{\,d} + 2n^2 P_2^{\,2d} + \frac{(4n)^3 P_2^{\,3d}}{1 - 4nP_2^{\,d}} \leq 8nP_2^{\,d} \rightarrow 0, \quad \text{if } d \sim \log n / Q.
\end{aligned}$$

Algorithm Average Complexity: Using Tarjan's parent pointer data representations for forests, we can find the optimal selection in $O(n \log n)$.

Multiple-choice Tries

Consider now k strings (before $k = 2$) per datum.

Consider now n independent vectors $\mathbf{X}_1, \ldots, \mathbf{X}_n$ of

$$k = \lceil c \log n \rceil, \quad c > 0$$

i.i.d. uniform $[0, 1]$ random variables $X_{i,j}, 1 \leq i \leq n, 1 \leq j \leq k$.

Theorem 5. *Let $\alpha \in (0, 1/3)$ and $c = 2/\alpha$. Then there exists a selection $Z_1, \ldots, Z_n$ such that the height H_n and fillup level F_n of the associated trie for*

$$X_{1,Z_1}, \ldots, X_{n,Z_n}$$

satisfy, for $n \geq 8$,

$$\mathbb{P}\{H_n - F_n \leq 2\} \geq 1 - \frac{3}{n}.$$

Thus $B_n = O(1)$ (existential result).

Greedy Heuristic (on-line algorithm) for $k = O(\log n)$ gives

$$H_n - F_n \leq 7, \quad \text{in probability}$$

that is, $B_n = O(1)$.

Analysis of Algorithms (AofA): Analytic Algorithmics

- **Analysis of Algorithms** is concerned with **precise** estimates of complexity parameters of algorithms and aims at predicting algorithms' behaviour. It develops general methods for obtaining closed-form formulae, asymptotic estimates, and probability distributions for combinatorial or probabilistic quantities. Properties of discrete structures such as strings, trees, tries, dags, graphs are investigated.

- The area of analysis of algorithms was born on July 27, 1963, when D. E. Knuth wrote his "Notes on Open Addressing".

- Following **Knuth and Hadamard's precept** [1], we study algorithmic problems using techniques of **complex analysis** such as generating functions, combinatorial calculus, Rice's formula, Mellin transform, Fourier series, sequences distributed modulo 1, saddle point methods, analytic poissonization and depoissonization, and singularity analysis.

- This program, which applies complex-analytic tools to analysis of algorithm, constitutes **analytic algorithmics**.

[1] The shortest path between two truths on the real line passes through the complex plane.


thank you!