# On Set Partitions, Words, Approximate Counting and Digital Search Trees

(joint with Chung-Kuei Lee and Helmut Prodinger)

Michael Fuchs

Institute of Applied Mathematics
National Chiao Tung University

Hsinchu, Taiwan

Changsha, June 28, 2013

# Set Partitions

**Example.**

$$\{\{2,7\},\{1,3,4\},\{5,6\}\}.$$

Set partition of $\{1,2,3,4,5,6,7\}$ with three blocks.

# Set Partitions

**Example.**

$$\{\{2,7\},\{1,3,4\},\{5,6\}\}.$$

Set partition of $\{1,2,3,4,5,6,7\}$ with three blocks.

$\#$ of set partitions of $\{1,\ldots,n\}$: **Bell number** $B_n$.

## Set Partitions

**Example.**

$$\{\{2,7\},\{1,3,4\},\{5,6\}\}.$$

Set partition of $\{1,2,3,4,5,6,7\}$ with three blocks.

$\#$ of set partitions of $\{1,\dots,n\}$: **Bell number** $B_n$.

We have,

$$B_n \sim n! \frac{e^{e^r-1}}{r^n\sqrt{2\pi r(r+1)e^r}},$$

where $re^r = n+1$, i.e., asymptotically

$$r = \log n - \log\log n + o(1)$$

# Number of Blocks

$\#$ of set partitions with $k$ blocks: **Stirling partition number** $S(n,k)$.

# Number of Blocks

\# of set partitions with $k$ blocks: **Stirling partition number** $S(n,k)$.

Assume that all partitions are equally likely.

# Number of Blocks

$\#$ of set partitions with $k$ blocks: **Stirling partition number** $S(n,k)$.

Assume that all partitions are equally likely.

$X_n = \#$ of blocks of a random partition. Then,

$$P(X_n = k) = \frac{S(n,k)}{B_n}.$$

# Number of Blocks

$\#$ of set partitions with $k$ blocks: **Stirling partition number** $S(n,k)$.

Assume that all partitions are equally likely.

$X_n = \#$ of blocks of a random partition. Then,

$$P(X_n = k) = \frac{S(n,k)}{B_n}.$$

**Theorem (Harper; 1967)**

*We have,*

$$\mathbb{E}(X_n) \sim \frac{n}{\log n}, \qquad \mathrm{Var}(X_n) \sim \frac{n}{\log^2 n}.$$

*Moreover,*

$$\frac{X_n - \mathbb{E}(X_n)}{\sqrt{\mathrm{Var}(X_n)}} \xrightarrow{d} N(0,1).$$

# Set Partitions as Words

Consider

$$\{\{2,7\},\{1,3,4\},\{5,6\}\}.$$

# Set Partitions as Words

Consider

$$\{\{2,7\},\{1,3,4\},\{5,6\}\}.$$

Choose smallest element of every block as block leader.

Arrange blocks such that block leaders are increasing.

# Set Partitions as Words

Consider

$$\{\{2,7\},\{1,3,4\},\{5,6\}\}.$$

Choose smallest element of every block as block leader.

Arrange blocks such that block leaders are increasing.

$$\{\overbrace{\{1,3,4\}}^{\#1},\overbrace{\{2,7\}}^{\#2},\overbrace{\{5,6\}}^{\#3}\}$$

## Set Partitions as Words

Consider

$$\{\{2,7\},\{1,3,4\},\{5,6\}\}.$$

Choose smallest element of every block as block leader.

Arrange blocks such that block leaders are increasing.

$$\{\overbrace{\{1,3,4\}}^{\#1},\overbrace{\{2,7\}}^{\#2},\overbrace{\{5,6\}}^{\#3}\}$$

$\omega_1 \cdots \omega_7$ with $w_i = \#$ of the block, i.e.,

$$\omega_1 \cdots \omega_7 = 1211332.$$

# Set Partitions as Words

Consider

$$\{\{2,7\},\{1,3,4\},\{5,6\}\}.$$

Choose smallest element of every block as block leader.

Arrange blocks such that block leaders are increasing.

$$\{\overbrace{\{1,3,4\}}^{\#1},\overbrace{\{2,7\}}^{\#2},\overbrace{\{5,6\}}^{\#3}\}$$

$\omega_1 \cdots \omega_7$ with $w_i = \#$ of the block, i.e.,

$$\omega_1 \cdots \omega_7 = 1211332.$$

This gives a 1-1 correspondence between set partitions and certain words.

# Geometric Words

$\omega = \omega_1 \cdots \omega_n$: word corresponding to a set partition.

# Geometric Words

$\omega = \omega_1 \cdots \omega_n$: word corresponding to a set partition.

$\omega$ satisfies **restricted growth property** (RGP):

$$\omega_i \leq 1 + \max\{\omega_0, \ldots, \omega_{i-1}\} \quad \text{with} \quad \omega_0 = 0.$$

# Geometric Words

$\omega = \omega_1 \cdots \omega_n$: word corresponding to a set partition.

$\omega$ satisfies **restricted growth property** (RGP):

$$\omega_i \le 1 + \max\{\omega_0, \ldots, \omega_{i-1}\} \quad \text{with} \quad \omega_0 = 0.$$

Largest letter of $\omega = \#$ of blocks.

# Geometric Words

$\omega = \omega_1 \cdots \omega_n$: word corresponding to a set partition.

$\omega$ satisfies **restricted growth property** (RGP):

$$\omega_i \leq 1 + \max\{\omega_0, \ldots, \omega_{i-1}\} \quad \text{with} \quad \omega_0 = 0.$$

Largest letter of $\omega$=# of blocks.

**Random Model on Words**:

$\omega_i$ independent, geometric random variables with success probability $p$.

## Geometric Words

$\omega = \omega_1 \cdots \omega_n$: word corresponding to a set partition.

$\omega$ satisfies **restricted growth property** (RGP):

$$\omega_i \leq 1 + \max\{\omega_0, \ldots, \omega_{i-1}\} \quad \text{with} \quad \omega_0 = 0.$$

Largest letter of $\omega$=# of blocks.

**Random Model on Words**:

$\omega_i$ independent, geometric random variables with success probability $p$.

Words generated by this random model are called **geometric words**.

# Geometric Words

$\omega = \omega_1 \cdots \omega_n$: word corresponding to a set partition.

$\omega$ satisfies **restricted growth property** (RGP):

$$\omega_i \leq 1 + \max\{\omega_0, \dots, \omega_{i-1}\} \quad \text{with} \quad \omega_0 = 0.$$

Largest letter of $\omega$=# of blocks.

**Random Model on Words**:

$\omega_i$ independent, geometric random variables with success probability $p$.

Words generated by this random model are called **geometric words**.

$p_n$: probability that a geometric word satisfies RGP.

## Results for $p_n$ (i)

$q = 1 - p$.

$(x;q)_n = (1-x)(1-xq)\cdots(1-xq^{n-1})$.

$(x;q)_\infty = \lim_{n\to\infty}(x;q)_n$.

**Theorem (Oliver, Prodinger; 2011; Mansour, Shattuck; 2012)**

*We have,*

$$\begin{aligned} p_n &= p\sum_{j=0}^{n-1}(-1)^j\binom{n-1}{j}q^j(p;q)_j \\ &= \sum_{j=0}^{n}(-1)^j\binom{n}{j}(p;q)_j. \end{aligned}$$

## Results for $p_n$ (ii)

$Q = 1/q$.

$L = \log Q$.

$\chi_k = 2\pi ik/L$.

**Theorem (Oliver, Prodinger; 2011)**

*We have,*

$$p_n \sim \frac{(p;q)_\infty}{L(q;q)_\infty}\Gamma(-\log_Q p)n^{\log_Q p} + n^{\log_Q p}\Psi(\log_Q n),$$

*where $\Psi(z)$ is the $1$-periodic function with average value $0$ and*

$$\Psi(z) = \frac{(p;q)_\infty}{L(q;q)_\infty}\sum_{k\neq 0}\Gamma(-\log_Q p + \chi_k)e^{-2\pi ikz}.$$

# Largest Letter

$p_{n,k}$: probability that a geometric word has largest letter $k$ and satisfies RGP.

# Largest Letter

$p_{n,k}$: probability that a geometric word has largest letter $k$ and satisfies RGP.

$$n_q = 1 + q + q^2 + \cdots + q^{n-1} = \frac{1-q^n}{1-q}.$$

$$n_q! = 1_q 2_q \cdots n_q.$$

$$\binom{n}{k}_q = \frac{n_q}{k_q (n-k)_q}.$$

**Theorem (Mansour, Shattuck; 2012)**

*We have,*

$$p_{n,k} = \frac{p^n}{k_q!} \sum_{j=0}^{k} (-1)^j q^{\binom{j}{2}} ((k-j)_q)^n \binom{k}{j}_q.$$

# Average Value of Largest Letter

$X_n$: largest letter of geometric word subject to RGP. Then,

$$P(X_n = k) = \frac{p_{n,k}}{p_n}.$$

## Average Value of Largest Letter

$X_n$: largest letter of geometric word subject to RGP. Then,

$$P(X_n = k) = \frac{p_{n,k}}{p_n}.$$

**Theorem (Prodinger; 2012)**

*We have,*

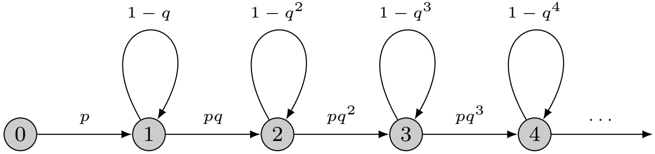
$$\mathbb{E}(X_n) \sim \log_Q n - \alpha_p - \frac{\psi(-\log_Q p)}{L} + \Phi(\log_Q n),$$

*where $\Phi(z)$ is a 1-periodic function with average value 0, $\psi = \Gamma'/\Gamma$ and*

$$\alpha_p = \sum_{l\geq 0} \frac{pq^l}{1 - pq^l}.$$

# Approximate Counting with Black Holes

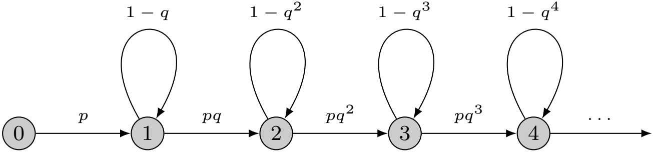
State diagram:




In every state there is a positive probability of violating RGP.

# Approximate Counting with Black Holes

State diagram:



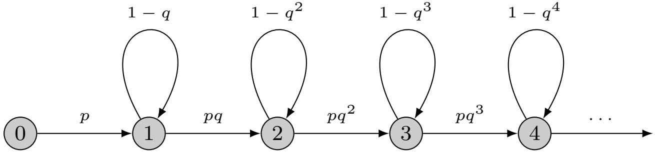

In every state there is a positive probability of violating RGP.

Above diagram implies

$$p_{n,k} = pq^{k-1}p_{n-1,k-1} + (1-q^k)p_{n-1,k}.$$

# Approximate Counting with Black Holes

State diagram:




In every state there is a positive probability of violating RGP.

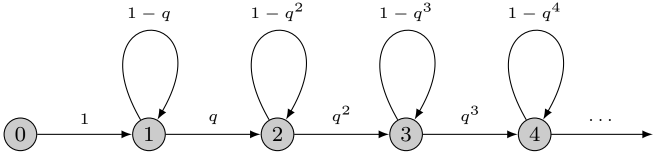
Above diagram implies

$$p_{n,k} = pq^{k-1}p_{n-1,k-1} + (1-q^k)p_{n-1,k}.$$

Prodinger used this as starting point for his analysis.

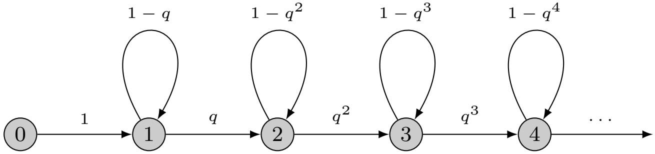
# Approximate Counting

State diagram:



# Approximate Counting

State diagram:

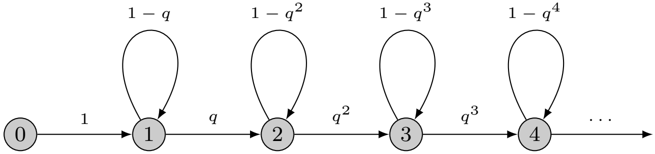



**Approximate Counting (Morris 1978)**:

Counter $C_n$ with $C_0 = 0$ and

$$C_{n+1} = \begin{cases} C_n + 1, & \text{with probability } q^{C_n}; \\ C_n, & \text{with probability } 1 - q^{C_n}. \end{cases}$$

State diagram:




**Approximate Counting (Morris 1978)**:

Counter $C_n$ with $C_0 = 0$ and

$$C_{n+1} = \begin{cases} C_n + 1, & \text{with probability } q^{C_n}; \\ C_n, & \text{with probability } 1 - q^{C_n}. \end{cases}$$

Only $\Theta(\log\log n)$ space is needed for counting $n$ objects.

# Applications

Approximate counting has found many applications:

- Analysis of the Webgraph.
- Monitoring network traffic.
- Finding patterns in protein and DNA sequencing.
- Computing frequency moments of data streams.
- Data storage in flash memory.
- Etc.

# Applications

Approximate counting has found many applications:

- Analysis of the Webgraph.
- Monitoring network traffic.
- Finding patterns in protein and DNA sequencing.
- Computing frequency moments of data streams.
- Data storage in flash memory.
- Etc.

Many refinements have been proposed.

# Analysis of Approximate Counting

**Flajolet (1985)**:

$$\mathbb{E}(C_n) \sim \log_Q n + C_{\text{mean}} + F(\log_Q n),$$

where $F(z)$ is a 1-periodic function

# Analysis of Approximate Counting

**Flajolet (1985)**:

$$\mathbb{E}(C_n) \sim \log_Q n + C_{\text{mean}} + F(\log_Q n),$$

where $F(z)$ is a 1-periodic function and

$$\text{Var}(C_n) \sim C_{\text{var}} + G(\log_Q n),$$

where $G(z)$ is a 1-periodic function and

$$C_{\text{var}} = \frac{\pi^2}{6L^2} - \alpha - \beta + \frac{1}{12} - \frac{1}{L} \sum_{l \geq 1} \frac{1}{l \sinh(2l\pi^2/L)}$$

with $\alpha = \sum_{l \geq 1} q^l/(1-q^l)$ and $\beta = \sum_{l \geq 1} q^{2l}/(1-q^l)^2$.

# Methods

Many different methods have been used:

- **Mellin Transform**: Flajolet (1985); Prodinger (1992)
- **Rice Method**: Kirschenhofer & Prodinger (1991)
- **Euler Transform**: Prodinger (1994)
- **Analysis of Extreme Value Distributions**: Louchard & Prodinger (2006)
- **Martingale Theory**: Rosenkrantz (1987)
- **Probability Theory**: Robert (2005)
- **Poisson-Laplace-Mellin Method**: F. & Lee & Prodinger (2012).

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

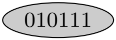
**Example:** a digital search tree build from $9$ keys:

010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:

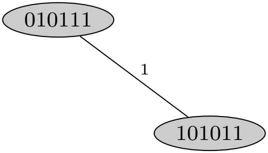



010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:

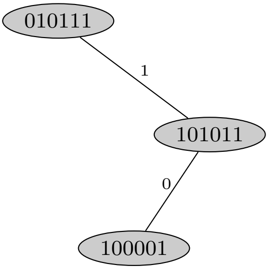



010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:

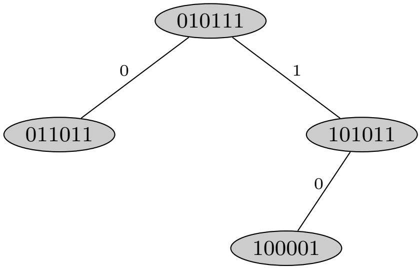



010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:

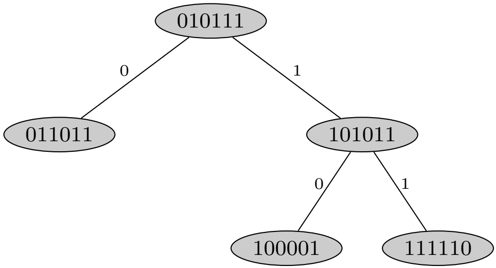



010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:

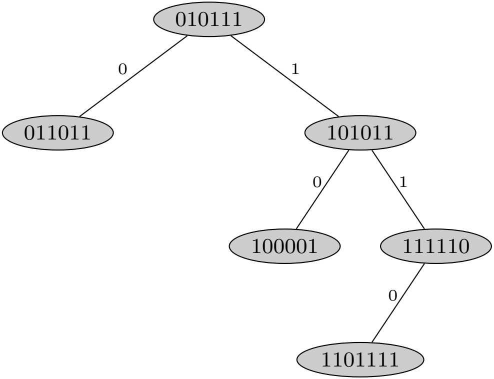



010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:

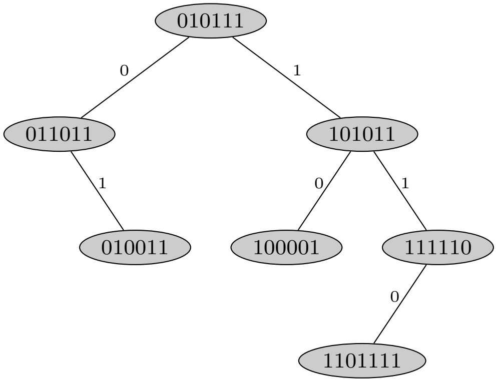



010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:

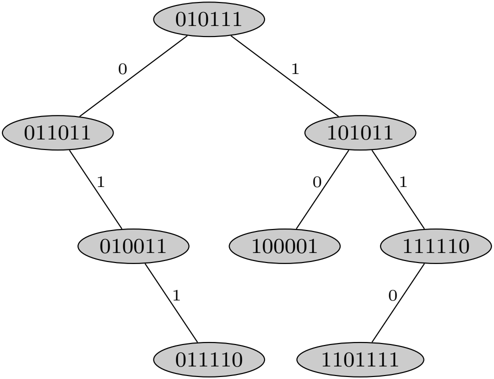



010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:

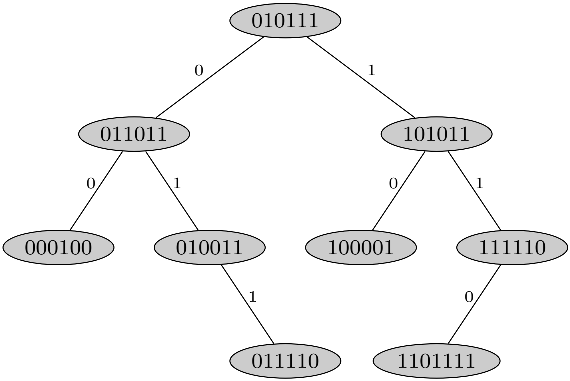



010111
101011
100001
011011
111110
110111
010011
011110
000100

# Digital Search Tree (DST)

Introduced by Coffman and Eve (1970).

**Example:** a digital search tree build from $9$ keys:




010111
101011
100001
011011
111110
110111
010011
011110
000100

# Random Model and Leftmost Path

**Random Model:**

Bits are generated by independent Bernoulli random variables with mean $p$.

# Random Model and Leftmost Path

**Random Model:**

Bits are generated by independent Bernoulli random variables with mean $p$.

Two types of trees:

- $p = 1/2$: *symmetric digital search tree*;
- $p \neq 1/2$: *asymmetric digital search tree*.

# Random Model and Leftmost Path

**Random Model:**

Bits are generated by independent Bernoulli random variables with mean $p$.

Two types of trees:

- $p = 1/2$: *symmetric digital search tree*;
- $p \neq 1/2$: *asymmetric digital search tree*.

**Length of the Leftmost Path**:

$X_n$: number of vertices on leftmost path.

# Random Model and Leftmost Path

**Random Model:**

Bits are generated by independent Bernoulli random variables with mean $p$.

Two types of trees:

- $p = 1/2$: *symmetric digital search tree*;
- $p \neq 1/2$: *asymmetric digital search tree*.

**Length of the Leftmost Path**:

$X_n$: number of vertices on leftmost path.

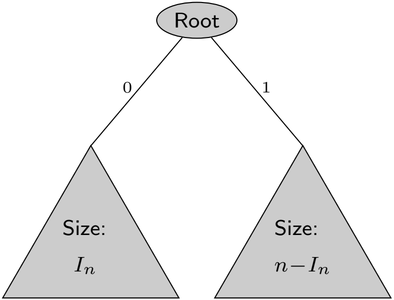
Note that:

$$X_n \stackrel{d}{=} C_n.$$

# Distributional Recurrence of $X_n$

$$X_{n+1} \stackrel{d}{=} X_{I_n} + 1$$

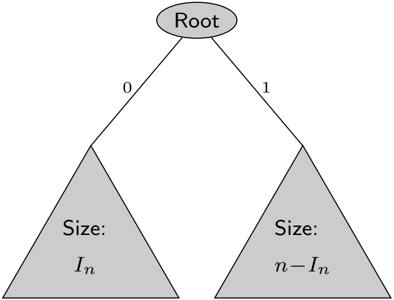
- $I_n \stackrel{d}{=}$ Binomial$(n, q)$;
- $X_n, I_n$ independent.



# Distributional Recurrence of $X_n$

$$X_{n+1} \stackrel{d}{=} X_{I_n} + 1$$

- $I_n \stackrel{d}{=} \text{Binomial}(n, q)$;
- $X_n, I_n$ independent.




Recurrence of moments:

$$f_{n+1} = \sum_{j=0}^{n} \binom{n}{j} q^j p^{n-j} f_j + g_n.$$

# Analytic Methods for DSTs

- **Rice Method**:

  Introduced by Flajolet and Sedgewick.

- **Approach of Flajolet and Richmond**:

  Based on Euler transform, Mellin transform, and singularity analysis.

- **Approach via Analytic Depoissonization**:

  Introduced by Jacquet & Regnier and Jacquet & Szpankowski. Based on saddle point method and Mellin transform.

- **Poisson-Laplace-Mellin Approach**:

  Introduced by F. & Hwang & Zacharovas. Based on analytic depoissonization and a combination of Laplace and Mellin transform.

## Variance of Approximate Counting

$Q_n = (q;q)_\infty/(q^{n+1};q)_\infty$; $Q_\infty = \lim_{n\to\infty} Q_n$.

**Theorem (F., Lee, Prodinger; 2012)**

*We have,*

$$\mathrm{Var}(C_n) \sim \sum_k g_k e^{2k\pi i \log_Q n},$$

*where*

$$g_k = \frac{Q_\infty}{L\Gamma(1+\chi_k)} \sum_{h,l,j\geq 0} \frac{(-1)^j q^{h+l+\binom{j+1}{2}}}{Q_h Q_l Q_j} \varphi(\chi_k; q^{h+j} + q^{l+j}).$$

*Here,*

$$\varphi(\chi; x) = \begin{cases} \pi(x^\chi - 1)/(\sin(\pi\chi)(x-1)), & \textit{if } x \neq 1; \\ \pi\chi/\sin(\pi\chi), & \textit{if } x = 1. \end{cases}$$

# An Identity

## Corollary (F., Lee, Prodinger; 2012)

*We have,*

$$\frac{Q_\infty}{L}\sum_{h,l,j\geq 0}\frac{(-1)^j q^{h+l+\binom{j+1}{2}}}{Q_h Q_l Q_j}\psi(q^{h+j}+q^{l+j})$$

$$=\frac{\pi^2}{6L^2}-\alpha-\beta+\frac{1}{12}-\frac{1}{L}\sum_{l\geq 1}\frac{1}{l\sinh(2l\pi^2/L)},$$

*where*

$$\psi(x)=\begin{cases}\log x/(x-1), & \textit{if } x\neq 1;\\ 1, & \textit{if } x=1.\end{cases}$$

# Geometric Words satisfying GRGP

$\omega = \omega_1 \cdots \omega_n$: geometric word.

# Geometric Words satisfying GRGP

$\omega = \omega_1 \cdots \omega_n$: geometric word.

$\omega$ satisfies **generalized restricted growth property** (GRGP):

$$\omega_i \le d + \max\{\omega_0, \ldots, \omega_{i-1}\} \quad \text{with} \quad \omega_0 = 0,$$

where $d \ge 1$ fixed.

## Geometric Words satisfying GRGP

$\omega = \omega_1 \cdots \omega_n$: geometric word.

$\omega$ satisfies **generalized restricted growth property** (GRGP):

$$\omega_i \leq d + \max\{\omega_0, \ldots, \omega_{i-1}\} \quad \text{with} \quad \omega_0 = 0,$$

where $d \geq 1$ fixed.

$p_n$: probability that a geometric word satisfies GRGP.

$p_{n,k}$: probability that a geometric word with largest letter $k$ satisfies GRGP.

$X_n$: largest letter of geometric word subject to GRGP. Again,

$$P(X_n = k) = \frac{p_{n,k}}{p_n}.$$

# Analysis of $p_n$ (i)

Conditioning on first letter and $\#$ of letters $\leq$ first letter:

$$p_{n+1} = \sum_{l=1}^{d} pq^{l-1} \sum_{j=0}^{n} \binom{n}{j} (1-q^l)^{n-j} q^{lj} p_j.$$

## Analysis of $p_n$ (i)

Conditioning on first letter and # of letters $\leq$ first letter:

$$p_{n+1} = \sum_{l=1}^{d} pq^{l-1} \sum_{j=0}^{n} \binom{n}{j} (1-q^l)^{n-j} q^{lj} p_j.$$

Set

$$\tilde{f}(z) = e^{-z} \sum_{n \geq 0} p_n \frac{z^n}{n!}.$$

# Analysis of $p_n$ (i)

Conditioning on first letter and # of letters $\leq$ first letter:

$$p_{n+1} = \sum_{l=1}^{d} pq^{l-1} \sum_{j=0}^{n} \binom{n}{j} (1-q^l)^{n-j} q^{lj} p_j.$$

Set

$$\tilde{f}(z) = e^{-z} \sum_{n\geq 0} p_n \frac{z^n}{n!}.$$

Then,

$$\tilde{f}(z) + \tilde{f}'(z) = \sum_{l=1}^{d} pq^{l-1} \tilde{f}(q^l z).$$

This is the probability in the **Poisson model**.

# Poisson Heuristic

**Poisson Heuristic:**

$$p_n \text{ sufficiently smooth} \quad \Longrightarrow \quad p_n \approx \tilde{f}(n) = e^{-n} \sum_{j\geq 0} p_j \frac{n^j}{j!}.$$

# Poisson Heuristic

**Poisson Heuristic:**

$$p_n \text{ sufficiently smooth} \quad \Longrightarrow \quad p_n \approx \tilde{f}(n) = e^{-n} \sum_{j\ge 0} p_j \frac{n^j}{j!}.$$

More precisely: if $p_n$ is smooth enough,

$$p_n \sim \sum_{j\ge 0} \frac{\tilde{f}^{(j)}(n)}{n!} \tau_j(n) = \tilde{f}(n) - \frac{n}{2} \tilde{f}''(n) + \dots,$$

where $\tau_j(n) := n![z^n](z-n)^j e^z$.

# Poisson Heuristic

**Poisson Heuristic:**

$$p_n \text{ sufficiently smooth} \quad \Longrightarrow \quad p_n \approx \tilde{f}(n) = e^{-n}\sum_{j\geq 0} p_j \frac{n^j}{j!}.$$

More precisely: if $p_n$ is smooth enough,

$$p_n \sim \sum_{j\geq 0} \frac{\tilde{f}^{(j)}(n)}{n!}\tau_j(n) = \tilde{f}(n) - \frac{n}{2}\tilde{f}''(n) + \dots,$$

where $\tau_j(n) := n![z^n](z-n)^j e^z$.

This is called *Poisson-Charlier expansion* (can be already found in Ramanujan's notebooks).

# Jacquet-Szpankowski-admissibility (JS-admissibility)

$\tilde{f}(z)$ is called JS-admissible if

$(\mathbf{I})$ Uniformly for $|\arg(z)| \le \epsilon$,

$$\tilde{f}(z) = \mathcal{O}\left(|z|^{\alpha} \log^{\beta}|z|\right),$$

$(\mathbf{O})$ Uniformly for $\epsilon < |\arg(z)| \le \pi$,

$$f(z) := e^{z}\tilde{f}(z) = \mathcal{O}\left(e^{(1-\epsilon)|z|}\right).$$

# Jacquet-Szpankowski-admissibility (JS-admissibility)

$\tilde{f}(z)$ is called JS-admissible if

**(I)** Uniformly for $|\arg(z)| \le \epsilon$,

$$\tilde{f}(z) = \mathcal{O}\left(|z|^{\alpha} \log^{\beta} |z|\right),$$

**(O)** Uniformly for $\epsilon < |\arg(z)| \le \pi$,

$$f(z) := e^{z}\tilde{f}(z) = \mathcal{O}\left(e^{(1-\epsilon)|z|}\right).$$

**Theorem (Jacquet, Szpankowski; 1998)**

*If $\tilde{f}(z)$ is JS-admissible, then*

$$f_n \sim \tilde{f}(n) - \frac{n}{2}\tilde{f}''(n) + \cdots .$$

# Depoissonization

JS-admissibility satisfies closure properties:

(i) $\tilde{f}, \tilde{g}$ JS-admissible, then $\tilde{f} + \tilde{g}$ JS-admissible.

(ii) $\tilde{f}$ JS-admissible, then $\tilde{f}'$ JS-admissible. Etc.

# Depoissonization

JS-admissibility satisfies closure properties:

(i) $\tilde{f}, \tilde{g}$ JS-admissible, then $\tilde{f} + \tilde{g}$ JS-admissible.

(ii) $\tilde{f}$ JS-admissible, then $\tilde{f}'$ JS-admissible. Etc.

# Depoissonization

JS-admissibility satisfies closure properties:

(i) $\tilde{f}, \tilde{g}$ JS-admissible, then $\tilde{f} + \tilde{g}$ JS-admissible.

(ii) $\tilde{f}$ JS-admissible, then $\tilde{f}'$ JS-admissible. Etc.

**Proposition**

*Consider*

$$\tilde{f}(z) + \tilde{f}'(z) = \sum_{l=1}^{d} pq^{l-1} \tilde{f}(q^l z) + \tilde{g}(z).$$

*We have,*

$$\tilde{g}(z) \text{ JS-admissible} \quad \Longleftrightarrow \quad \tilde{f}(z) \text{ JS-admissible.}$$

# Analysis of $p_n$ (ii)

Recall

$$\tilde{f}(z) + \tilde{f}'(z) = \sum_{l=1}^{d} pq^{l-1}\tilde{f}(q^{l}z).$$

# Analysis of $p_n$ (ii)

Recall

$$\tilde{f}(z) + \tilde{f}'(z) = \sum_{l=1}^{d} pq^{l-1}\tilde{f}(q^{l}z).$$

Obviously, $\tilde{f}(z)$ is JS-admissible. Thus,

$$p_n \sim \tilde{f}(n).$$

We only have to find an asymptotic of $\tilde{f}(z)$.

# Analysis of $p_n$ (ii)

Recall

$$\tilde{f}(z) + \tilde{f}'(z) = \sum_{l=1}^{d} pq^{l-1} \tilde{f}(q^{l}z).$$

Obviously, $\tilde{f}(z)$ is JS-admissible. Thus,

$$p_n \sim \tilde{f}(n).$$

We only have to find an asymptotic of $\tilde{f}(z)$.

This can be done via Mellin transform.

$$\mathscr{M}[\tilde{f}(z); s] = \int_0^\infty \tilde{f}(z) z^{s-1} \mathrm{d}z.$$

# Analysis of $p_n$ (iii)

We have,

$$\mathscr{M}[\tilde{f}(z); s] = \frac{q^d \Omega(1)\Gamma(s)}{P(q^{-s})\Omega(q^{-s})},$$

where

$$P(z) = 1 - p\sum_{l=1}^{d} q^{l-1} z^l$$

and

$$\Omega(s) = \prod_{j\geq 1} P(sq^j).$$

## Analysis of $p_n$ (iii)

We have,

$$\mathscr{M}[\tilde{f}(z); s] = \frac{q^d \Omega(1)\Gamma(s)}{P(q^{-s})\Omega(q^{-s})},$$

where

$$P(z) = 1 - p \sum_{l=1}^{d} q^{l-1} z^l$$

and

$$\Omega(s) = \prod_{j \geq 1} P(sq^j).$$

**Lemma**

*Let $\rho$ be the smallest positive root of $P(z)$. Then, $\rho$ is simple and the only root with $|z| \leq \rho$.*

## Converse Mapping Theorem

**Theorem (Flajolet, Gourdon, Dumas; 1995)**

*Let the Mellin transform of $\tilde{f}(z)$ exist in the strip $\langle \alpha, \beta \rangle$.*

*Assume that $\mathscr{M}[\tilde{f}(z); s]$ can be continued to a meromorphic function on $\langle \alpha, \gamma \rangle$ with $\beta < \gamma$ with simple poles at $s_1, \cdots, s_k$.*

*Then, under some technical conditions,*

$$\tilde{f}(z) = -\sum_{j=1}^{k} \mathrm{Res}(\mathscr{M}[\tilde{f}(z); s], s = s_j) z^{-s_j} + \mathcal{O}\left(z^{-\gamma}\right)$$

*as $z \to \infty$.*

# Analysis of $p_n$ (iv)

$\mathscr{M}[\tilde{f}(z);s]$ has simple poles at $\log_Q \rho + \chi_k$ with

$$\mathrm{Res}(\mathscr{M}[\tilde{f}(z);s]) = \frac{q^d \Omega(1)}{L\rho P'(\rho)\Omega(\rho)} \Gamma(\log_Q \rho + \chi_k).$$

# Analysis of $p_n$ (iv)

$\mathscr{M}[\tilde{f}(z); s]$ has simple poles at $\log_Q \rho + \chi_k$ with

$$\mathrm{Res}(\mathscr{M}[\tilde{f}(z); s]) = \frac{q^d \Omega(1)}{L\rho P'(\rho)\Omega(\rho)} \Gamma(\log_Q \rho + \chi_k).$$

Thus,

$$\tilde{f}(z) \sim -\frac{q^d \Omega(1)}{L\rho P'(\rho)\Omega(\rho)} z^{-\log_Q \rho} \sum_k \Gamma(\log_Q \rho + \chi_k) z^{-\chi_k}$$

and

$$p_n \sim -\frac{q^d \Omega(1)}{L\rho P'(\rho)\Omega(\rho)} n^{-\log_Q \rho} \sum_k \Gamma(\log_Q \rho + \chi_k) n^{-\chi_k}.$$

# Result for $p_n$

**Theorem (F., Prodinger; 2013)**

*We have,*

$$p_n \sim -\frac{q^d\Omega(1)}{L\rho P'(\rho)\Omega(\rho)}\Gamma(\log_Q \rho) n^{-\log_Q \rho} + n^{-\log_Q \rho}\Psi(\log_Q n),$$

*where $\Psi(z)$ is the $1$-periodic function with average value $0$ and*

$$\Psi(z) = -\frac{q^d\Omega(1)}{L\rho P'(\rho)\Omega(\rho)}\sum_{k\neq 0}\Gamma(\log_Q \rho + \chi_k)e^{-2\pi i k z}.$$

For $d = 1$: $\rho = 1/p$ and result coincides with Oliver and Prodinger's result.

## Average Value of $X_n$

$X_n$: largest letter of geometric word subject to GRGP.

## Average Value of $X_n$

$X_n$: largest letter of geometric word subject to GRGP.

Similar (but more involved) analysis gives:

**Theorem (F., Prodinger; 2013)**

*We have,*

$$\mathbb{E}(X_n) \sim \log_Q n - \alpha_p - \frac{\psi(\log_Q \rho)}{L} + \Phi(\log_Q n),$$

*where $\Phi(z)$ is a 1-periodic function with average value 0, $\psi = \Gamma'/\Gamma$ and*

$$\alpha_p = -\sum_{l \geq 0} \frac{q^l P'(q^l)}{P(q^l)}.$$

## Further Extensions

- **Further Restrictions on Geometric Words**:

  Geometric words satisfying RGP with largest letter $k$ and fixed levels, rises, descends, etc.

# Further Extensions

- **Further Restrictions on Geometric Words**:

  Geometric words satisfying RGP with largest letter $k$ and fixed levels, rises, descends, etc.

- **More properties of $X_n$**:

  Find variance, higher moments and limit laws.

# Further Extensions

- **Further Restrictions on Geometric Words**:

  Geometric words satisfying RGP with largest letter $k$ and fixed levels, rises, descends, etc.

- **More properties of $X_n$**:

  Find variance, higher moments and limit laws.

- **Generality of our method**:

  The method seems to be applicable to asymmetric DSTs with $\log p / \log q \in \mathbb{Q}$. This might yield simplifications of expressions in asymptotics of total path length, peripheral path length, profile, number of leaves, patterns, etc.