String Complexity

Wojciech Szpankowski
Purdue University
W. Lafayette, IN 47907

June 1, 2015

Center for
Science of Information
NSF Science & Technology Center

Dedicated to Svante Janson for his 60 Birthday

Outline

1. Working with Svante
2. String Complexity
3. Joint String Complexity

Joint Papers

1. S. Janson and W. Szpankowski, Analysis of an asymmetric leader election algorithm *Electronic J. of Combinatorics*, 4, R17, 1997.

2. S. Janson, S. Lonardi, and W. Szpankowski, **On Average Sequence Complexity**, *Theoretical Computer Science*, 326, 213–227, 2004 (also *Combinatorial Pattern Matching* Conference, CPM'04, Istanbul, 2004).

3. S. Janson and W. Szpankowski, Partial Fillup and Search Time in LC Tries *ACM Trans. on Algorithms*, 3, 44:1-44:14, 2007 (also, *ANALCO*, Miami, 2006).

4. A. Magner, S. Janson, G. Kollias, and W. Szpankowski On Symmetry of Uniform and Preferential Attachment Graphs, *Electronic J. Combinatorics*, 21, P3.32, 2014 (also, *25th International Conference on Probabilistic, Combinatorial and Asymptotic Methods for the Analysis of Algorithms* AofA'14, Paris, 2014).

Joint Papers

1. S. Janson and W. Szpankowski, Analysis of an asymmetric leader election algorithm *Electronic J. of Combinatorics*, 4, R17, 1997.

2. S. Janson, S. Lonardi, and W. Szpankowski, **On Average Sequence Complexity**, *Theoretical Computer Science*, 326, 213–227, 2004 (also *Combinatorial Pattern Matching* Conference, CPM'04, Istanbul, 2004).

3. S. Janson and W. Szpankowski, Partial Fillup and Search Time in LC Tries *ACM Trans. on Algorithms*, 3, 44:1-44:14, 2007 (also, *ANALCO*, Miami, 2006).

4. A. Magner, S. Janson, G. Kollias, and W. Szpankowski On Symmetry of Uniform and Preferential Attachment Graphs, *Electronic J. Combinatorics*, 21, P3.32, 2014 (also, *25th International Conference on Probabilistic, Combinatorial and Asymptotic Methods for the Analysis of Algorithms* AofA'14, Paris, 2014).

Working with Svante is easy ....


HAWAII
SWEDEN

Outline

1. Working with Svante
2. String Complexity
3. Joint String Complexity

Some Definitions

String Complexity of a single sequence is the number of distinct substrings.

Throughout, we write X for the string and denote by $I(X)$ the set of *distinct* substrings of X over alphabet $\mathcal{A}$.

Example. If $X = aabaa$, then

$$I(X) = \{\epsilon, a, b, aa, ab, ba, aab, aba, baa, aaba, abaa, aabaa\},$$

and $|I(X)| = 12$. But if $X = aaaaa$, then

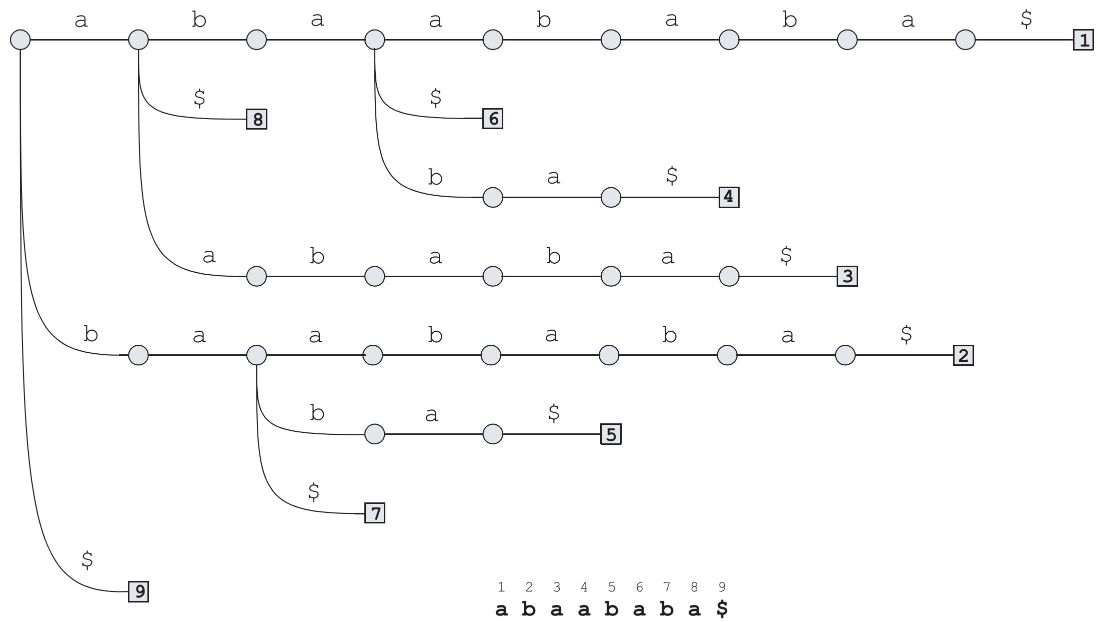
$$I(X) = \{\epsilon, a, aa, aaa, aaaa, aaaaa\},$$

and $|I(X)| = 6$.

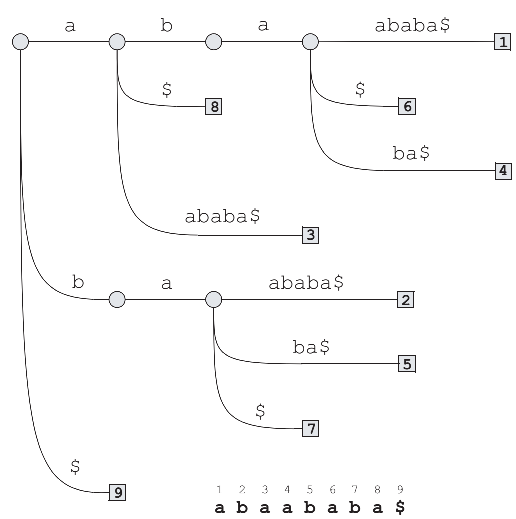
The string complexity is the cardinality of $I(X)$ and we study here the *average* string complexity.

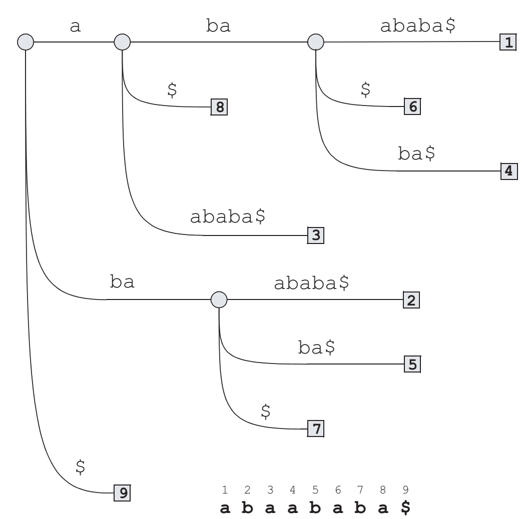
$$\mathbf{E}[|I(X)|] = \sum_{X \in \mathcal{A}^n} P(X)|I(X)|.$$

Suffix Trees and String Complexity



Non-compact suffix trie for $X = \texttt{abaababa}$ and string complexity $I(X) = 24$.





String Complexity = # internal nodes in a non-compact suffix tree.

Some Simple Facts

Let $O(w)$ denote the number of times that the word w occurs in X. Then

$$|I(X)| = \sum_{w \in \mathcal{A}^*} \min\{1, O(w)\}.$$

Since between any two positions in X there is one and only one substring:

$$\sum_{w \in \mathcal{A}^*} O(w) = \frac{(|X|+1)|X|}{2}.$$

Hence

$$|I(X)| = \frac{(|X|+1)|X|}{2} - \sum_{w \in \mathcal{A}^*} \max\{0, O(w) - 1\}.$$

Define: $C_n := \mathbf{E}[|I(X)| \mid |X| = n]$. Then

$$C_n = \frac{(n+1)n}{2} - \sum_{w \in \mathcal{A}^*} \sum_{k \geq 2} (k-1) P(O_n(w) = k).$$

We need to study probabilistically $O_n(w)$: that is:

$\boxed{\text{number of } w \text{ occurrences in a text } X \text{ generated a probabilistic source.}}$

New Book on Pattern Matching

How do you distinguish a cat from a dog by their DNA?
Did Shakespeare really write all of his plays?

Pattern matching techniques can offer answers to these questions and to many others, from molecular biology, to telecommunications, to classifying Twitter content.

This book for researchers and graduate students demonstrates the probabilistic approach to pattern matching, which predicts the performance of pattern matching algorithms with very high precision using analytic combinatorics and analytic information theory. Part I compiles known results of pattern matching problems via analytic methods. Part II focuses on applications to various data structures on words, such as digital trees, suffix trees, string complexity and string-based data compression. The authors use results and techniques from Part I and also introduce new methodology such as the Mellin transform and analytic depoissonization.

More than 100 end-of-chapter problems help the reader to make the link between theory and practice.

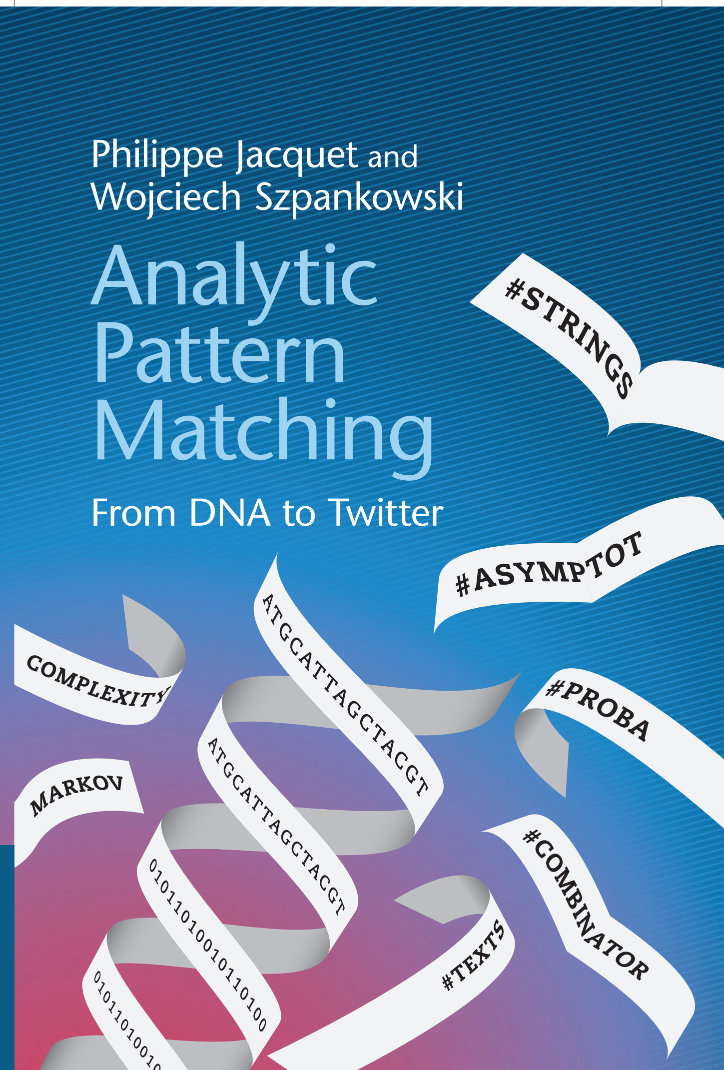
Philippe Jacquet is a research director at INRIA, a major public research lab in Computer Science in France. He has been a major contributor to the Internet OLSR protocol for mobile networks. His research interests involve information theory, probability theory, quantum telecommunication, protocol design, performance evaluation and optimization, and the analysis of algorithms. Since 2012 he has been with Alcatel-Lucent Bell Labs as head of the department of Mathematics of Dynamic Networks and Information. Jacquet is a member of the prestigious French Corps des Mines, known for excellence in French industry, with the rank of "Ingenieur General". He is also a member of ACM and IEEE.

Wojciech Szpankowski is Saul Rosen Professor of Computer Science and (by courtesy) Electrical and Computer Engineering at Purdue University, where he teaches and conducts research in analysis of algorithms, information theory, bioinformatics, analytic combinatorics, random structures, and stability problems of distributed systems. In 2008 he launched the interdisciplinary Institute for Science of Information, and in 2010 he became the Director of the newly established NSF Science and Technology Center for Science of Information. Szpankowski is a Fellow of IEEE and an Erskine Fellow. He received the Humboldt Research Award in 2010.

Cover design: Andrew Ward

CAMBRIDGE
UNIVERSITY PRESS
www.cambridge.org
ISBN 978-0-521-87608-7

Jacquet and Szpankowski Analytic Pattern Matching CAMBRIDGE



Book Contents

Some Results

Last expression allows us to write

$$C_n = \frac{(n+1)n}{2} + \mathbf{E}[S_n] - \mathbf{E}[L_n]$$

where $\mathbf{E}[S_n]$ and $\mathbf{E}[L_n]$ are, respectively, the average size and path length in the associated (compact) suffix trees.

We know that

$$\begin{aligned} \mathbf{E}[S_n] &= \frac{1}{h}(n + \Psi(\log n)) + o(n), \\ \mathbf{E}[L_n] &= \frac{n \log n}{h} + n\Psi_2(\log n) + o(n), \end{aligned}$$

where $\Psi(\log n)$ and $\Psi_2(\log n)$ are periodic functions (when the $\log p_a$, $a \in \mathcal{A}$ are rationally related), and where h is the entropy rate. Therefore,

$$C_n = \frac{(n+1)n}{2} - \frac{n}{h}(\log n - 1 + Q_0(\log n) + o(1))$$

where $Q_0(x)$ is a periodic function.

Theorem from 2004 Proved with Bare-Hands

In 2004 Svante, Stefano and I published the first result of this kind for a symmetric memoryless source (all symbol probabilities are the same).

Theorem 1 (Janson, Lonardi, W.S., 2004). *Let C_n be the string complexity for an unbiased memoryless source over alphabet $\mathcal{A}$. Then*

$$\mathbf{E}(C_n) = \binom{n+1}{2} - n \log_{|\mathcal{A}|} n + \left(\frac{1}{2} + \frac{1-\gamma}{\ln |\mathcal{A}|} + \varphi_{|\mathcal{A}|}(\log_{|\mathcal{A}|} n) \right) n + O(\sqrt{n \log n})$$

where $\gamma \approx 0.577$ is Euler's constant and

$$\varphi_{|\mathcal{A}|}(x) = -\frac{1}{\ln |\mathcal{A}|} \sum_{j \neq 0} \Gamma\left(-1 - \frac{2\pi i j}{\ln |\mathcal{A}|}\right) e^{2\pi i j x}$$

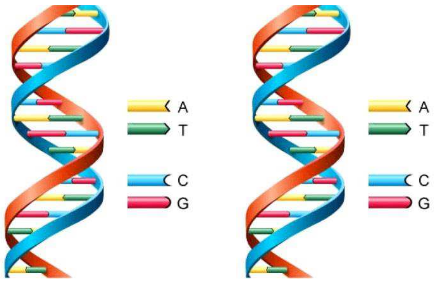
is a continuous function with period 1. $|\varphi_{|\mathcal{A}|}(x)|$ is very small for small $|\mathcal{A}|$: $|\varphi_2(x)| < 2 \cdot 10^{-7}$, $|\varphi_3(x)| < 5 \cdot 10^{-5}$, $|\varphi_4(x)| < 3 \cdot 10^{-4}$.

Outline

1. Working with Svante
2. String Complexity
3. Joint String Complexity

Joint String Complexity

For X and Y, let $J(X,Y)$ be the set of common words between X and Y.



The joint string complexity is
$|J(X,Y)| = |I(X) \cap I(Y)|$

Example. If $X = aabaa$ and $Y = abbba$, then $J(X,Y) = \{\varepsilon, a, b, ab, ba\}$.

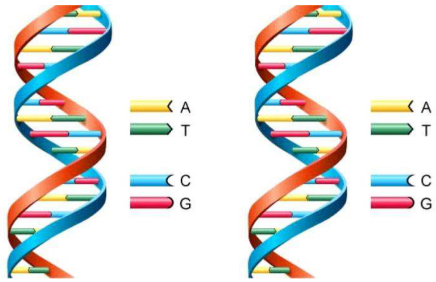
Goal. Estimate

$$J_{n,m} = \mathbf{E}[|J(X,Y)|]$$

when $|X| = n$ and $|Y| = m$.

Joint String Complexity

For X and Y, let $J(X,Y)$ be the set of common words between X and Y.



The joint string complexity is
$|J(X,Y)| = |I(X) \cap I(Y)|$

Example. If $X = aabaa$ and $Y = abbba$, then $J(X,Y) = \{\varepsilon, a, b, ab, ba\}$.

Goal. Estimate

$$J_{n,m} = \mathbf{E}[|J(X,Y)|]$$

when $|X| = n$ and $|Y| = m$.

Some Observations. For any word $w \in \mathcal{A}^*$

$$|J(X,Y)| = \sum_{w \in \mathcal{A}^*} \min\{1, O_X(w)\} \cdot \min\{1, O_Y(w)\}.$$

When $|X| = n$ and $|Y| = m$, we have

$$J_{n,m} = \mathbf{E}[|J(X,Y)|] - 1 = \sum_{w \in \mathcal{A}^* - \{\varepsilon\}} P(O_n^1(w) \geq 1) P(O_m^2(w) \geq 1)$$

where $O_n^i(w)$ is the number of w-occurrences in a string of generated by source $i = 1, 2$ (i.e., X and Y) which we assume to be memoryless sources.

Independent Joint String Complexity

Consider two sets of n independently generated (memoryless) strings.

Let $\Omega_n^i(w)$ be the number of strings for which w is a **prefix** when the n strings are generated by a source $i = 1, 2$ define

$$C_{n,m} = \sum_{w \in \mathcal{A}^* - \{\varepsilon\}} P(\Omega_n^1(w) \geq 1) P(\Omega_m^2(w) \geq 1)$$

Theorem 2. *There exists $\varepsilon > 0$ such that*

$$J_{n,m} - C_{n,m} = O(\min\{n, m\}^{-\varepsilon})$$

for large n.

Independent Joint String Complexity

Consider two sets of n independently generated (memoryless) strings.

Let $\Omega_n^i(w)$ be the number of strings for which w is a **prefix** when the n strings are generated by a source $i = 1, 2$ define

$$C_{n,m} = \sum_{w \in \mathcal{A}^* - \{\varepsilon\}} P(\Omega_n^1(w) \geq 1) P(\Omega_m^2(w) \geq 1)$$

Theorem 2. *There exists $\varepsilon > 0$ such that*

$$J_{n,m} - C_{n,m} = O(\min\{n, m\}^{-\varepsilon})$$

for large n.

Recurrence for $C_{n,m}$

$$C_{n,m} = 1 + \sum_{a \in \mathcal{A}} \sum_{k,\ell \geq 0} \binom{n}{k} P_1(a)^k (1 - P_1(a))^{n-k} \binom{m}{\ell} P_2(a)^\ell (1 - P_2(a))^{m-\ell} C_{k,\ell}$$

with $C_{0,m} = C_{n,0} = 0$.

Poisson Transform. The Poisson transform $C(z_1, z_2)$ of $C_{n,m}$ is

$$C(z_1, z_2) = \sum_{n,m \geq 0} C_{n,m} \frac{z_1^n z_2^m}{n!m!} e^{-z_1 - z_2}.$$

which becomes the functional equation after summing up the recurrence:

$$C(z_1, z_2) = (1 - e^{-z_1})(1 - e^{-z_2}) + \sum_{a \in \mathcal{A}} C\left(\mathrm{P}_1(a) z_1, \mathrm{P}_2(a) z_2\right).$$

Clearly, $n!m!C_{n,m} = [z_1^n][z_2^m]C(z_1, z_2)e^{z_1 + z_2}$.

Generating Functions, Mellin Transform, DePoissonization . . .

Poisson Transform. The Poisson transform $C(z_1, z_2)$ of $C_{n,m}$ is

$$C(z_1, z_2) = \sum_{n,m\geq 0} C_{n,m} \frac{z_1^n z_2^m}{n!m!} e^{-z_1 - z_2}.$$

which becomes the functional equation after summing up the recurrence:

$$C(z_1, z_2) = (1 - e^{-z_1})(1 - e^{-z_2}) + \sum_{a\in\mathcal{A}} C\left(\mathrm{P}_1(a) z_1, \mathrm{P}_2(a) z_2\right).$$

Clearly, $n!m!C_{n,m} = [z_1^n][z_2^m]C(z_1, z_2)e^{z_1+z_2}$.

Mellin Transform. Two dimensional Mellin transform is defined as

$$C^*(s_1, s_2) = \int_0^\infty \int_0^\infty C(z_1, z_2) z_1^{s_1-1} z_2^{s_2-1} dz_1 dz_2.$$

From the above functional equation we find for $-2 < \Re(s_i) < -1$

$$C^*(s_1, s_2) = \Gamma(s_1)\Gamma(s_2)\left(\frac{1}{H(s_1, s_2)} + \frac{s_1}{H(-1, s_2)} + \frac{s_2}{H(s_1, -1)} + \frac{s_1 s_2}{H(-1, -1)}\right)$$

where the kernel is defined as

$$H(s_1, s_2) = 1 - \sum_{a\in\mathcal{A}} (P_1(a))^{-s_1}(P_2(a))^{-s_2}.$$

Finding $C_{n,n}$

Here we only consider $m = n$ and $z_1 = z_2 = z$.

To recover $C_{n,n}$ we first find the inverse Mellin

$$C(z, z) = \frac{1}{(2i\pi)^2} \int_{\Re(s_1)=c_1} \int_{\Re(s_2)=c_2} C^*(s_1, s_2) z^{-s_1-s_2} ds_1 ds_2$$

which turns out to be

$$C(z, z) = \left(\frac{1}{2i\pi}\right)^2 \int_{\Re(s_1)=\rho_1} \int_{\Re(s_2)=\rho_2} \frac{\Gamma(s_1)\Gamma(s_2)}{H(s_1, s_2)} z^{-s_1-s_2} ds_1 ds_2 + o(z^{-M}),$$

for any $M > 0$ as $z \to \infty$ in a cone around the real axis.

The final step to recover

$$C_{n,n} \sim C(n, n)$$

is to apply the two-dimensional depoissonization.

Main Results

Assume that $\forall a \in \mathcal{A}$ we have $P_1(a) = P_2(a) = p_a$.

Theorem 3. *For a biased memoryless source, the joint complexity is asymptotically*

$$C_{n,n} = n\frac{2\log 2}{h} + Q(\log n)n + o(n),$$

where $Q(x)$ is a small periodic function (with amplitude smaller than 10^{-6}) which is nonzero only when the $\log p_a$, $a \in \mathcal{A}$, are rationally related, that is, $\log p_a / \log p_b \in \mathbb{Q}$.

Assume that $P_1(a) \neq P_2(a)$.

Theorem 4. *Define $\kappa = \min_{(s_1,s_2)\in\mathcal{K}\cap\mathbb{R}^2}\{(-s_1 - s_2)\} < 1$, where s_1 and s_2 are roots of*

$$H(s_1, s_2) = 1 - \sum_{a\in\mathcal{A}} (P_1(a))^{-s_1}(P_2(a))^{-s_2} = 0.$$

Then

$$C_{n,n} = \frac{n^{\kappa}}{\sqrt{\log n}}\left(\frac{\Gamma(c_1)\Gamma(c_2)}{\sqrt{\pi\Delta H(c_1, c_2)\nabla H(c_1, c_2)}} + Q(\log n) + O(1/\log n)\right),$$

where Q is a double periodic function.

Very Brief Sketch of Proof

1. Set $P_1(a) = 1/|\mathcal{A}|$ and then the kernel is

$$H(s_1, s_2) = 1 - |\mathcal{A}|^{s_1} \sum_{a \in \mathcal{A}} p_a^{s_2}.$$

Define $r(s_2) = \sum_{a \in \mathcal{A}} p_a^{s_2}$ and $L(s_2) = \log_{|\mathcal{A}|} r(s_2)$.

2. Roots of $H(s_1, s_2) = 0$ are

$$s_1 = -\log_{|\mathcal{A}|}(r(s_2)) + \frac{2ik\pi}{\log(|\mathcal{A}|)}$$

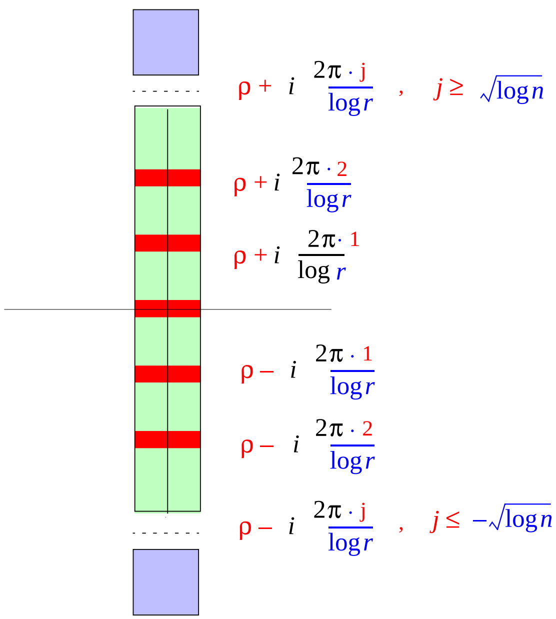
which are poles of $C(z, z)$ leading to

$$C(z, z) \sim \frac{1}{2i\pi \log|\mathcal{A}|} \int_{\Re(s)=c_2} \sum_k \Gamma\left(-L(s) + \frac{2ik\pi}{\log(|\mathcal{A}|)}\right) \Gamma(s) z^{L(s)-s-2ik\pi/\log(|\mathcal{A}|)} ds$$

Integrating over $s = s_2$ requires the **saddle point** method.

Saddle Point

3. The function $L(s) - s$ achieves it minimum at $c_2 =: \rho$ is the dominant real saddle point. But there is more . . .



Infinitely Many Saddle Points:

3a. $L(c_2 + it)$ is a periodic function with period $2\pi \log \nu$.

3b The saddle points are at $c_2 + 2\pi i\ell / \log \nu$.

3c. The infinite saddle points defines the fluctuating function Q.

4. The growth of $C(z, z)$ is defined by $\quad z^{L(c_2)-c_2} = z^{\kappa} \quad$ where

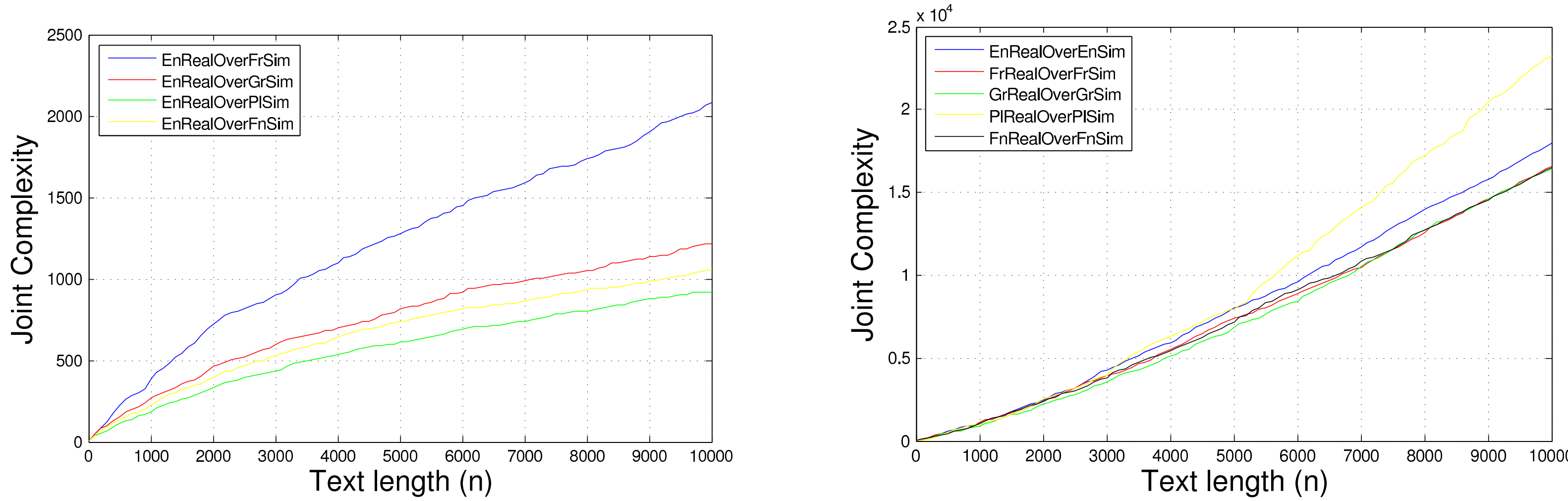
$$\kappa = \min_{s \in \mathbb{R}} \{\log_{|\mathcal{A}|}(r(s)) - s\}, \qquad c_2 = \min \arg_{s \in \mathbb{R}} \{\log_{|\mathcal{A}|}(r(s)) - s\},$$

where here $s = s_2$, and recall $L(s_2) = \log_{|\mathcal{A}|} r(s_2)$.
The factor $1/\sqrt{\log n}$ comes from the saddle point approximation. This completes the sketch.

Classification of Sources

The growth of $C_{n,n}$ is:

- $\Theta(n)$ for identical sources;
- $\Theta(n^{\kappa}/\sqrt{\log n})$ for nonidential sources with $\kappa < 1$.



(a) (b)

Figure 1: Joint complexity: (a) English text vs French, Greek, Polish, and Finnish texts; (b) real and simulated texts (3rd Markov order) of English, French, Greek, Polish and Finnish language.

That's It

THANK YOU, SVANTE!