Kolmogorov structure functions for automatic complexity

Bjørn Kjos-Hanssen

June 16, 2015 – *Varieties of Algorithmic Information*, University of Heidelberg – Internationales Wissenschaftssentrum

History

- 1936: Universal Turing machine / programming language / computer
- 1965: Kolmogorov complexity of a string $x = 0111001$, say, $=$ the length of the shortest program printing x
- 1973: Structure function (Kolmogorov)
- 2001: Automatic complexity $A(x)$ (Shallit and Wang) (deterministic)
- 2013: Nondeterministic automatic complexity $A_N(x)$ (Hyde, M.A. thesis, University of Hawai'i): $n/2$ upper bound.
- 2014: Structure function for automatic complexity: entropy-based upper bound.

Definition of A_N, nondeterministic automatic complexity

- Let M be an nondeterministic finite automaton having q states. If there is **exactly one path through M of length $|x|$ leading to an accept state**, and x is the string read along the path, then we say $A_N(x) \leq q$.
- Such an NFA is said to witness the complexity of x being no more than q.

Upper Bound

Theorem 1 (Hyde 2013)

Let $|\Sigma| \geq 2$ *be fixed and suppose* $x \in \Sigma^n$*. Then* $A_N(x) \leq \frac{n}{2} + 1$*.*

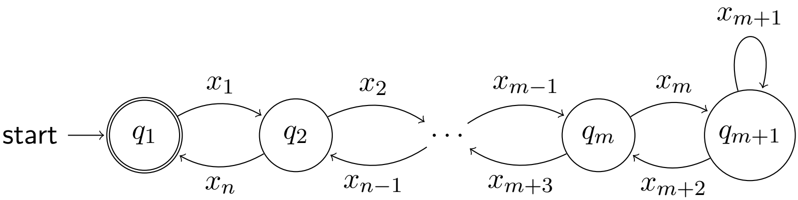

Figure 1: An NFA uniquely accepting $x = x_1 x_2 \ldots x_n$, $n = 2m + 1$

Example

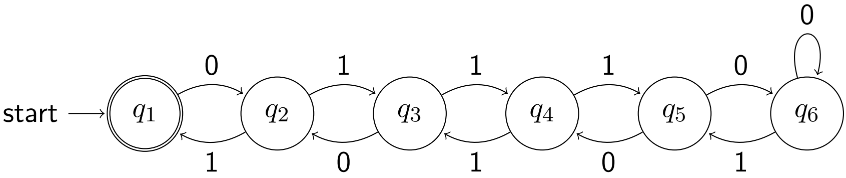

Figure 2: An NFA uniquely accepting $x = 01110010101$, $|x| = 11$, $A_N(x) \leq 6$.

Background

The Kolmogorov complexity of a finite word w is roughly speaking the length of the shortest description w^* of w in a fixed formal language. The description w^* can be thought of as an optimally compressed version of w. Motivated by the non-computability of Kolmogorov complexity, Shallit and Wang studied a deterministic finite automaton analogue.

Definition 1 (Shallit and Wang 2001)

The automatic complexity of a finite binary string $x = x_1 \dots x_n$ *is the least number* $A_D(x)$ *of states of a deterministic finite automaton* M *such that* x *is the only string of length* n *in the language accepted by* M*.*

Deficiency

Definition 2

The deficiency of a word x of length n is

$$D_n(x) = D(x) = b(n) - A_N(x).$$

where $b(n) = \lfloor \frac{n}{2} \rfloor + 1$ is Hyde's upper bound.

Length n	$\mathbb{P}(D_n > 0)$
0	0.000
2	0.500
4	0.500
6	0.531
8	0.617
10	0.664
12	0.600
14	0.687
16	0.657
18	0.658
20	0.641
22	0.633
24	0.593

(a) Even lengths.

Length n	$\mathbb{P}(D_n > 0)$
1	0.000
3	0.250
5	0.250
7	0.234
9	0.207
11	0.317
13	0.295
15	0.297
17	0.342
19	0.330
21	0.303
23	0.322
25	0.283

(b) Odd lengths.

Table 1: Probability of strings of having positive complexity deficiency D_n, truncated to 3 decimal digits.

Definition 3

Let $\mathrm{DEFICIENCY}$ *be the following decision problem.*
Given a binary word w *and an integer* $d \geq 0$*, is* $D(w) > d$*?*

Theorem 4

$\mathrm{DEFICIENCY}$ *is in NP.*

We do not know whether $\mathrm{DEFICIENCY}$ is NP-complete.
Can you find deficiencies? Play the Complexity Guessing Game or the Complexity Option Game.

The next table shows the number of strings of length $0 \leq n \leq 15$ having nondeterministic automatic complexity k. (Paper contains a table for $n \leq 23$.) We note that each column has a limiting value (2,6,20,58,...).

$n\backslash k$	1	2	3	4	5	6	7	8
15	2	6	20	58	226	908	8530	23018
14	2	6	20	58	244	1270	9668	5116
13	2	6	20	64	250	2076	5774	
12	2	6	20	58	282	2090	1638	
11	2	6	20	58	564	1398		
10	2	6	20	64	588	344		
9	2	6	20	78	406			
8	2	6	20	130	98			
7	2	6	22	98				
6	2	6	26	30				
5	2	6	24					
4	2	6	8					
3	2	6						
2	2	2						
1	2							
0	1							

Structure function

Let

$$S_x = \{(q,m) \mid \exists\ q\text{-state NFA } M, x \in L(M) \cap \Sigma^n, |L(M) \cap \Sigma^n| \le b^m\}$$

where $b = |\Sigma|$. Then S_x has the upward closure property

$$q \le q', m \le m', (q,m) \in S_x \quad \Longrightarrow \quad (q', m') \in S_x.$$

Structure function

Definition 5 (Vereshchagin, personal communication, 2014)

In an alphabet Σ containing b symbols, we define

$$h_x^*(m) = \min\{k : (k,m) \in S_x\} \quad \textit{and}$$

$$h_x(k) = \min\{m : (k,m) \in S_x\}.$$

Thus $(k,m) \in S_x$ iff $h_x^*(m) \leq k$ iff $h_x(k) \leq m$.

Example 6

The string $x = 01111111101111111$ *has an "optimal explanation" given by an automaton which goes to a new state when reading a 0, and stays in the same state when reading a 1. In this case* $h_x(3)$ *is more unusually small (the* p*-value associated with* $h_x(3)$ *is small) than other* $h_x(k)$*.*

Definition 7

The entropy function $\mathcal{H} : [0,1] \to [0,1]$ *is given by*

$$\mathcal{H}(p) = -p \log_2 p - (1-p)\log_2(1-p).$$

A useful well-known fact:

Theorem 8

For $0 \leq k \leq n$*,*

$$\log_2 \binom{n}{k} = \mathcal{H}(k/n)n + O(\log n).$$

Definition 9

The dual automatic structure function of a string x of length n is a function $h_x^* : [0, n] \to [0, \lfloor n/2 \rfloor + 1]$. We define the asymptotic upper envelope of h^* by

$$\widetilde{h^*}(a) = \limsup_{n\to\infty} \max_{|x|=n} \frac{h_x^*([a \cdot n])}{n}, \quad \widetilde{h^*} : [0, 1] \to [0, 1/2]$$

Let

$$\tilde{h}(p) = \limsup_{n\to\infty} \max_{|x|=n} \frac{h_x([p \cdot n])}{n}, \quad \tilde{h} : [0, 1/2] \to [0, 1]$$

where $[x]$ is the nearest integer to x.

We are interested in upper bounds on $\tilde{h}$.

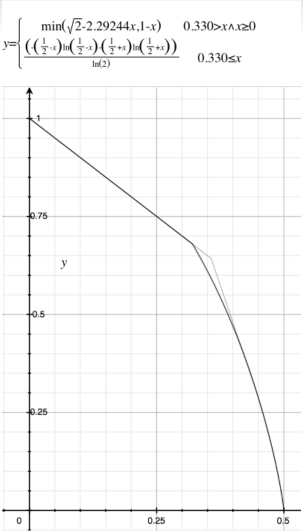

Figure 3: Bounds for the automatic structure function for alphabet size $b = 2$.

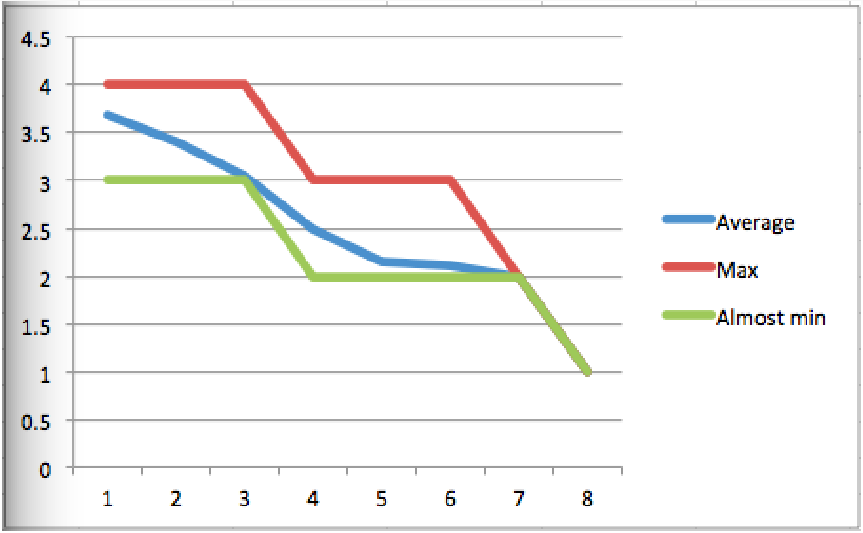

Figure 4: Automatic structure function for alphabet size $b = 2$ and length 7.

Multirun complexity

A *multirun* is a disjoint collections of runs of varying valences. For instance, in 011111023232 the most unlikely may be the two runs 11111 and 23232.
This corresponds to automata with no back-tracking.

Question 10

Given a word over a k-ary alphabet, can we efficiently find the multirun which has smallest p-value?

There are too many sequences of possible runs to consider them all, but perhaps there is another way.
Structure Function Calculator