

# Automaticity

Jeffrey Shallit  
Department of Computer Science  
University of Waterloo  
Waterloo, Ontario N2L 3G1  
Canada

`shallit@graceland.uwaterloo.ca`  
`http://math.uwaterloo.ca/~shallit`

This talk represents joint work with Y. Breitbart, I. Glaister, J. M. Robson, and C. Pomerance.

## **Problem:**

Given an object (finite string, infinite string, language, etc.), assign a measure of its complexity.

## **One Solution:**

- Kolmogorov–(Chaitin–Solomonoff) complexity
  - See, for example, the recent book by Ming Li and Paul Vitányi
- “The complexity of an object is the length of the shortest program to produce it.”
- very powerful idea
- lots of applications
- unfortunately not computable!
- is there a computable analogue?

## Automaticity

- idea: like Kolmogorov complexity, but replace “Turing machine” with finite automaton
- goals:
  - measure complexity of languages
  - complexity measure is a function
  - regular languages should have  $O(1)$  automaticity
  - languages “close” to regular should have “small” automaticity

## Basics of Finite Automata

- If  $\Sigma$  is a finite set of symbols, then by  $\Sigma^*$  we mean the free monoid over  $\Sigma$  (set of all finite strings of symbols chosen from  $\Sigma$ );
- A *language* is a subset of  $\Sigma^*$ .
- a *finite automaton* is a simple model of a computer
- formally it is a quintuple:  $M = (Q, \Sigma, \delta, q_0, F)$  where:
  - $Q$  is a finite set of *states*;
  - the *size* of  $M$  is  $|M| := |Q|$ , the number of states;
  - $\Sigma$  is a finite set of symbols, called the *input alphabet*;
  - $q_0 \in Q$  is the *start state*;
  - $F \subseteq Q$  is the set of *final states*;
  - $\delta : Q \times \Sigma \rightarrow Q$  is the *transition function*
- The *language accepted by*  $M$  is denoted by  $L(M)$  and is given by  $\{w \in \Sigma^* \mid \delta(q_0, w) \in F\}$ .

## Automaticity Defined

- $\Sigma^{\leq n} = \epsilon + \Sigma + \Sigma^2 + \dots + \Sigma^{\leq n}$ , the set of all strings in  $\Sigma^*$  of length  $\leq n$ .
- a language  $L \subseteq \Sigma^*$  is an  $n$ 'th order approximation to a language  $L'$  if  $L \cap \Sigma^{\leq n} = L' \cap \Sigma^{\leq n}$ .
- automaticity of a language  $L$  is the function which counts the number of states in the smallest DFA that accepts some  $n$ 'th order approximation to  $L$
- DFA = class of all deterministic (complete) finite automata over a finite alphabet  $\Sigma$
- Formally, the function  $A_L(n)$  is the automaticity of the language  $L$ , where  $A_L(n) = \min\{|M| : M \in \text{DFA and } L(M) \cap \Sigma^{\leq n} = L \cap \Sigma^{\leq n}\}$ .

## Basic Properties of Automaticity

1.  $A_L(n) \leq A_L(n + 1)$ .
2.  $L$  is regular iff  $A_L(n) = O(1)$ .
3.  $A_L(n) = A_{\overline{L}}(n)$ .
4.  $A_L(n) \leq 2 + \sum_{w \in L \cap \Sigma^{\leq n}} |w|$ .

## An Example

Consider the language

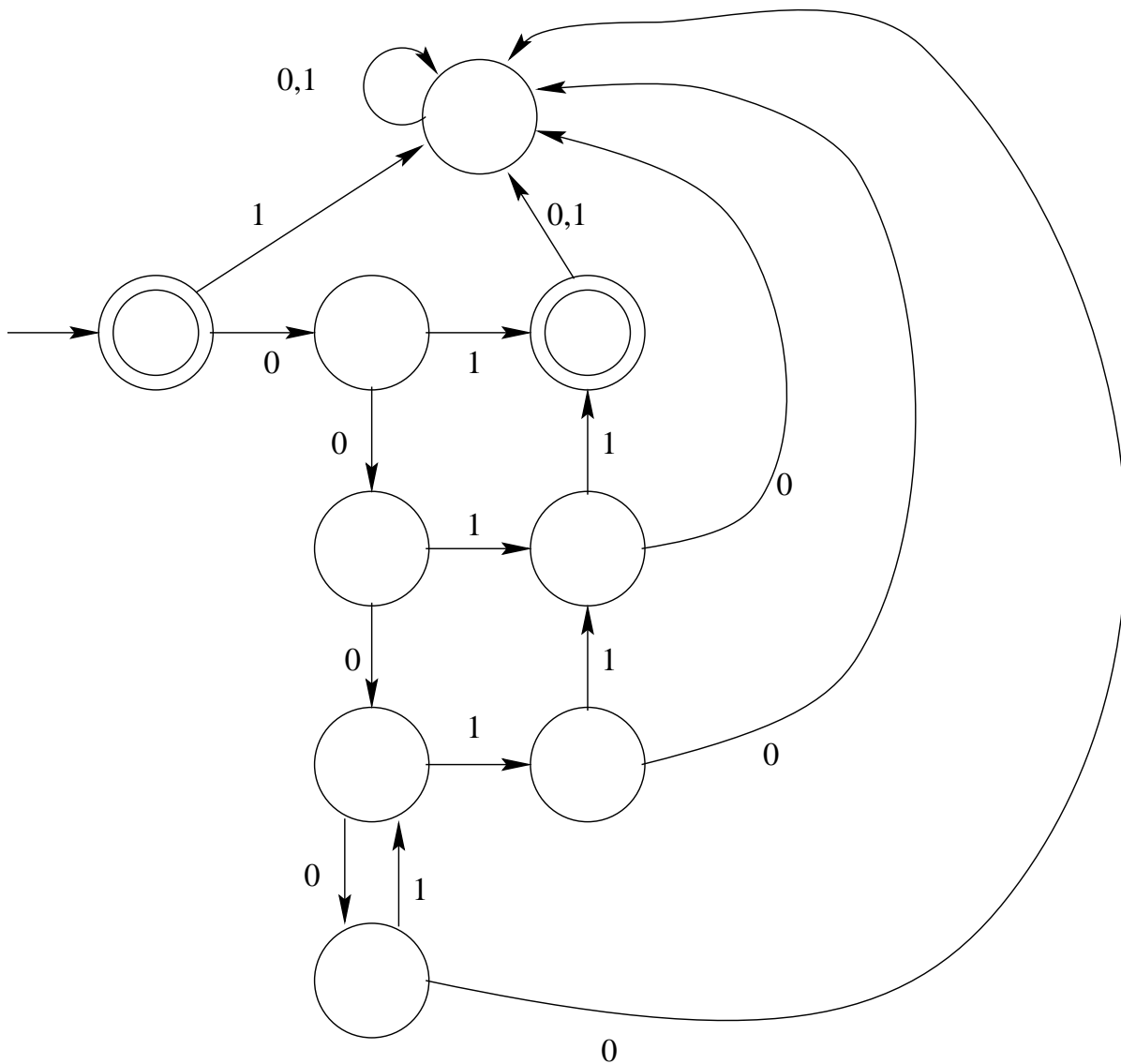
$$L = \{0^n 1^n : n \geq 0\}.$$

This language is clearly not regular. What is its automaticity?

## Automaticity of $L = \{0^n 1^n : n \geq 0\}$

The automaticity of  $L$  is  $A_L(n) = 2\lfloor n/2 \rfloor + 1$  for  $n \geq 2$ .

To see the upper bound, note that we can accept an  $n$ 'th order approximation to  $L$  (for  $n = 9$ ) with the following DFA:





## Another Example

Consider

$$P = \{10, 11, 101, 111, 1011, 1101, 10001, 10011, \dots\},$$

the set of primes expressed in base-2.

A classical theorem due to Minsky and Papert (1966) shows that  $P$  is not a regular language.

**Theorem.** The automaticity of  $P^R$  is  $\Omega(2^{n/43})$ .

(Here  $P^R$  denotes the reversal of the set  $P$ , i.e. the primes expressed with least significant digit first.)

The proof depends on a deep theorem of Heath-Brown on the distribution of small primes in arithmetic progressions.

## Previous Work

- Trakhtenbrot (1964)
  - called automaticity the “weight of a finite tree”
  - used a different model of automaton
  - proved upper and lower bounds
- Grinberg & Korshunov (1966)
  - improved bounds of Trakhtenbrot
- Karp (1967)
  - introduced (but did not name) the concept of automaticity exactly as we defined it
  - proved that if  $L$  is not a regular language, then  $A_L(n) \geq (n + 3)/2$  for infinitely many  $n$
  - observed that the constants 2 and 3 in this theorem are best possible (but proof was flawed)

- Breitbart (1970, 1971, 1973)
  - studied the automaticity of the language  $L_{k,p} = \{w : w \text{ is the base-}p \text{ representation of a perfect } k\text{'th power}\}$ .
  - gave examples of languages with high automaticity
- Paredaens & Vyncke (1977); Gabarró (1983); Balcázar, Díaz, & Gabarró (1985); Serna (1989)
  - studied similar notions
- Dwork and Stockmeyer (1989)
  - introduced “measure of nonregularity”
  - the words  $w, w'$  are called  $n$ -dissimilar if  $|w|, |w'| \leq n$  and there exists a word  $v$  with  $|wv|, |w'v| \leq n$  and  $wv \in L$  iff  $w'v \notin L$ .
  - set  $A'_L(n) =$  the maximum number of distinct pairwise  $n$ -dissimilar words
  - proved: if a language  $L$  is recognized by a 2PFA with probability  $1/2 + \delta$  ( $\delta > 0$ ) in time  $c^{n^{o(1)}}$ , then  $L$  is regular.

- Kaneps & Freivalds (1990)

- They defined  $r_{\text{sim}}(L, \Sigma^{\leq n})$  = same definition as Dwork & Stockmeyer for  $n$ -dissimilar words
- $r_{\text{sim}}(L, \Sigma^{\leq n}) = A_L(n)$
- independently proved Karp's theorem
- proved: if a language  $L$  is recognized by a probabilistic TM with probability  $1/2 + \delta$  for some  $\delta > 0$  in  $O(\log \log n)$ , then  $L$  is regular

## Bounds on Automaticity

**Theorem.** Let  $k = |\Sigma|$  and  $C = 1/((\log_2 k)(k - 1)^2)$ .  
Then

$$A_L(n) \leq \frac{Ck^{n+2}}{n}(1 + o(1)).$$

**Theorem.** Let  $k = |\Sigma|$ . Then for “almost all” languages  $L \subseteq \Sigma^*$ , we have

$$A_L(n) > (1 - \epsilon) \frac{Ck^{n+1}}{n}$$

for any fixed  $\epsilon > 0$  and all sufficiently large  $n$ .

## Karp's Theorem

**Theorem.** Let  $L \subseteq \Sigma^*$  be a nonregular language. Then

$$A_L(n) \geq (n + 3)/2$$

for infinitely many  $n$ .

**Theorem.** The constants 3 and 2 in Karp's theorem are best possible, in the sense that the theorem would be false if 2 were replaced with any smaller number, or if 3 were replaced with any larger number.

## Automaticity: The Unary Case

- $k = |\Sigma| = 1$
- different issues arise
- we have  $A_L(n) \leq n + 1$ , for all  $L$  and for all  $n$ .

**Theorem.** Let  $L \subseteq 0^*$ . Then

$$A_L(n) \leq n + 1 - \lfloor \log_2 n \rfloor$$

for infinitely many  $n$ .

**Theorem.** Let  $L \subseteq 0^*$ . Then for “almost all”  $L$  we have

$$A_L(n) > n - 2 \log_2 n - 2 \log_2 \log_2 n$$

for all sufficiently large  $n$ .

## Can Karp's Theorem be Improved in the Unary Case?

Recall that Karp proved that if  $L$  is not regular, then  $A_L(n) \geq (n + 3)/2$  infinitely often. This implies that

$$\limsup_{n \rightarrow \infty} \frac{A_L(n)}{n} \geq \frac{1}{2}$$

for all nonregular  $L$ .

**Conjecture.** If  $L \subseteq 0^*$  is not regular, then

$$\limsup_{n \rightarrow \infty} \frac{A_L(n)}{n} \geq \frac{\sqrt{5} - 1}{2} \doteq .61803.$$

This conjecture can be expressed in the following equivalent formulation:



## An Equivalent Conjecture

**Conjecture.** Let  $w = w_0w_1w_2 \cdots$  be an infinite word over a finite alphabet that is not ultimately periodic. Define  $s_w(n)$  to be the length of the longest suffix of

$$w_0w_1w_2 \cdots w_n$$

that also appears as a subword of  $w_0w_1w_2 \cdots w_{n-1}$ . Then

$$\liminf_{n \rightarrow \infty} \frac{s_w(n)}{n} \leq \frac{3 - \sqrt{5}}{2} \doteq .38197.$$

The upper bound is attained for the infinite Fibonacci word (discussed below).

This conjecture has been open since 1994. Recently, E. Garel obtained some partial results.

## The Fibonacci Language

Define  $h_1 = 1$ ,  $h_2 = 0$ , and  $h_{n+1} = h_n h_{n-1}$  for  $n \geq 2$ .

Then

$$h_3 = 01$$

$$h_4 = 010$$

$$h_5 = 01001$$

$$h_6 = 01001010$$

$$h_7 = 0100101001001$$

⋮

Now define

$$\lim_{n \rightarrow \infty} h_n = f = f_0 f_1 f_2 \dots;$$

this is the infinite “Fibonacci word”. Let

$$L_f = \{0^i : f_i = 0\} = \{\epsilon, 0^2, 0^3, 0^5, 0^7, 0^8, \dots\}.$$

**Theorem.**

$$\limsup_{n \rightarrow \infty} \frac{A_{L_f}(n)}{n} = \frac{\sqrt{5} - 1}{2}.$$

## Nondeterministic Automaticity

Let NFA = class of all nondeterministic finite automata.

The function  $N_L(n)$  is the nondeterministic automaticity of the language  $L$ , where  $N_L(n) = \min\{|M| : M \in \text{NFA} \text{ and } L(M) \cap \Sigma^{\leq n} = L \cap \Sigma^{\leq n}\}$ .

**Theorem.** Let  $k = |\Sigma| \geq 2$  and  $L \subseteq \Sigma^*$ . Then  $N_L(n) = O(k^{n/2})$ .

**Theorem.** Let  $k = |\Sigma| \geq 2$ , and let  $L \subseteq \Sigma^*$ . Then for almost all  $L$  and every  $\epsilon > 0$  we have

$$N_L(n) > (1 - \epsilon)k^{n/2} / \sqrt{k - 1}$$

for all sufficiently large  $n$ .

## Lower Bounds for Nondeterministic Automaticity

**Theorem.** Suppose  $L \subseteq \Sigma^*$ . If  $L$  is not regular, then  $N_L(n) \geq \log_2((n + 3)/2)$  for infinitely many  $n$ .

This lower bound is best possible, up to a constant, since the Stearns-Hartmanis-Lewis language

$$\{2 (1)_2^R 2 (2)_2^R 2 (3)_2^R 2 (4)_2^R 2 \cdots 2 (n)_2^R : n \geq 1\}$$

has nondeterministic automaticity  $O(\log n)$ . Here  $(k)_2$  is the representation of  $k$  in base-2, and  $w^R$  denotes the reversal of the string  $w$ .

## Another Example of a Language with Low Nondeterministic Automaticity

**Theorem.** Define

$$L = \{w \in (0 + 1)^* : |w|_0 \neq |w|_1\}.$$

Then  $L$  is nonregular and

$$N_L(n) = O((\log n)^2 / (\log \log n)).$$

*Proof.* We need the following fact from number theory:

Let  $n \geq 2$  and suppose  $0 \leq i, j < n$ . Then  $i \neq j$  iff there exists a prime  $p \leq 4.4 \log n$  such that  $i \not\equiv j \pmod{p}$ .

Thus, to nondeterministically accept some  $n$ th order approximation to  $L$ , we can

- “guess” the correct prime  $p \leq 4.4 \log n$ ;
- “verify” that  $|w|_0 \not\equiv |w|_1 \pmod{p}$ .

This construction uses at most

$$1 + \sum_{p \leq 4.4 \log n} p = O((\log n)^2 / (\log \log n))$$

states.

## A General Method for Lower Bounds for Nondeterministic Automaticity

**Theorem.** Let  $L$  be a language, and suppose there exists a set of pairs  $P = \{(x_i, w_i) : 1 \leq i \leq m\}$  such that

(a)  $x_i w_i \in L \cup \Sigma^{\leq n}$ ; and

(b)  $x_j w_i \notin L$  for  $1 \leq i, j \leq m, i \neq j$ , and  $|x_j w_i| \leq n$ .

Then  $N_L(n) \geq m$ .

*Proof.* Let  $M = (Q, \Sigma, \delta, q_0, F)$  be any nondeterministic finite automaton accepting an  $n$ th order approximation to  $L$ . Now  $x_i w_i \in L$ , and  $|x_i w_i| \leq n$ . Since  $M$  accepts all strings of length  $\leq n$  in  $L$ , we have  $\delta(q_0, x_i w_i) \cup F \neq \emptyset$ . Hence there exists at least one state  $q \in \delta(q_0, x_i)$  such that  $p \in \delta(q, w_i)$ , where  $p \in F$ .

However, for every *other* string  $w_j, (j \neq i)$ , we must have  $q \notin \delta(q_0, x_j)$ . For if  $q \in \delta(q_0, x_j)$ , we would have  $p \in \delta(q_0, x_j w_i)$  and so  $x_j w_i \in L$ , a contradiction, since  $|x_j w_i| \leq n$ .

Hence every set  $\delta(q_0, x_i)$  contains a state  $q$  which does not appear in any other set  $\delta(q_0, x_j)$  with  $j \neq i$ . It follows that there must be at least  $m$  different states in  $M$ .

## Lower Bounds for Nondeterm. Automaticity The Unary Case

**Theorem.** There exists a constant  $c$  (which does not depend on  $L$ ) such that if  $L \subseteq 0^*$  is not regular, then

$$N_L(n) \geq c(\log n)^2 / (\log \log n)$$

infinitely often.

**Open Question:** Is this lower bound best possible?

Carl Pomerance has recently shown that for all monotonically increasing functions  $f$ , there exists a language  $L = L(f)$  such that

$$N_L(n) = O(f(n)(\log n)^2 / (\log \log n)),$$

thus showing the lower bound is essentially tight.

## Low Nondeterministic Automaticity: A Unary Example

**Theorem.** Define  $L = \{0^n : n \geq 1 \text{ and the least positive integer not dividing } n \text{ is not a power of } 2\}$ . Then  $L$  is nonregular and

$$N_L(n) = O((\log n)^3 / (\log \log n)).$$

*Proof.* The construction depends on the following two facts:

If  $0^n \in L$ , then there exists a prime power  $p^k$ ,  $p \geq 3$ ,  $k \geq 1$ ,  $p^k \leq 5 \log n$ , such that  $n \not\equiv 0 \pmod{p^k}$ , and  $n \equiv 0 \pmod{2^s}$ , with  $2^s < p^k < 2^{s+1}$ .

Further, if such a prime power  $p^k$  exists, then  $0^n \in L$ .

An NFA accepting an  $n$ -th order approximation to  $L$  can now be constructed as follows:



## Low Nondeterministic Automaticity: A Unary Example

- “guess” the correct odd prime power  $p^k \leq 5 \log n$ ;
- verify that, on input  $0^r$ , we have
  - \*  $r \not\equiv 0 \pmod{p^k}$
  - \*  $r \equiv 0 \pmod{2^s}$ , with  $2^s < p^k < 2^{s+1}$ .

This construction uses at most  $O((\log n)^3 / (\log \log n))$  states.

## Polynomial Automaticity

Define the following three complexity classes:

- deterministic polynomial automaticity, or DPA

$$\text{DPA} = \{L \subseteq \Sigma^* : \exists k \text{ such that } A_L(n) = O(n^k)\}.$$

- nondeterministic polynomial automaticity, or NPA

$$\text{NPA} = \{L \subseteq \Sigma^* : \exists k \text{ such that } N_L(n) = O(n^k)\}.$$

- nondeterministic poly-log automaticity, or NPLA

$$\text{NPLA} = \{L \subseteq \Sigma^* : \exists k \text{ such that } N_L(n) = O((\log n)^k)\}.$$

What are the closure properties of these classes, and how are they related?

## A Hierarchy of Polynomial Automaticity

**Theorem.** For all integers  $k \geq 0$  there exists a language  $L_k$  such that  $A_{L_k} = \Theta(n^k)$ .

*Proof.* Let

$$L_k = \{0^{a_1} 1 0^{a_2} 1 \dots 0^{a_k} 1 \ 0^{a_1} 1 0^{a_2} 1 \dots 0^{a_k} 1 : \\ a_1, a_2, \dots, a_k \geq 0\}.$$

**Theorem.** The class DPA is closed under union, intersection, complement, and inverse homomorphism.

*Proof.* Simply adapt the usual constructions.

**Theorem.** The class DPA is not closed under concatenation.

*Proof.* Let  $L_1 = (0+1)^*$  and  $L_2 = \{1(0+1)^{2^k} : k \geq 0\}$ . Then  $A_{L_1}(n) = O(1)$ , and  $A_{L_2}(n) = O(n)$ , but it can be shown that  $A_{L_1 L_2}(n) \geq 2^{n/3}$  for infinitely many  $n$ .

## An Open Question

**Open Question.** Is  $\text{NPLA} \subseteq \text{DPA}$ ?

## For Further Reading

1. J. Shallit and Y. Breitbart, Automaticity: Properties of a measure of descriptive complexity, in *STACS '94*, Lecture Notes in Comp. Sci. # 775, pp. 619–630. Revised version, to appear, *J. Comput. System Sci.*
2. C. Pomerance and J. M. Robson and J. Shallit, Automaticity II: Descriptive complexity in the unary case, submitted.
3. I. Glaister and J. Shallit, Automaticity III: Polynomial automaticity and context-free languages, to appear, Proceedings of MFCS '96.
4. J. Shallit, Automaticity IV: Sequences, Sets, and Diversity, to appear, *J. de Théorie des Nombres de Bordeaux*.