Finite State Transducers with Intuition*

Ruben Agadzanyan and Rūsiņš Freivalds

Institute of Mathematics and Computer Science, University of Latvia,
Raiņa bulvāris 29, Riga, LV-1459, Latvia

Abstract. Finite automata that take advice have been studied from the point of view of what is the amount of advice needed to recognize nonregular languages. It turns out that there can be at least two different types of advice. In this paper we concentrate on cases when the given advice contains zero information about the input word and the language to be recognized. Nonetheless some nonregular languages can be recognized in this way. The help-word is merely a sufficiently long word with nearly maximum Kolmogorov complexity. Moreover, any sufficiently long word with nearly maximum Kolmogorov complexity can serve as a help-word. Finite automata with such help can recognize languages not recognizable by nondeterministic nor probabilistic automata. We hope that mechanisms like the one considered in this paper may be useful to construct a mathematical model for human intuition.

1 Introduction

The main difficulty in all the attempts to construct a mathematical model for human intuition has been the failure to explain the nature of the "outside help from nowhere" that characterizes our understanding of this (maybe, nonexistant) phenomenon. If this help contains a piece of information about my problem to be solved then it is unbelievable that somebody sends such a help to me. If this help does not contain information then why this message can help me.

We propose one possible mechanism of such a help based on finite automata that take an advice and other nonconstructive methods of computation. The help indeed contains zero information about the problem but nontheless it helps. Moreover, it turns out that this mechanism is different from probabilistic, nondeterministic and quantum computation.

The use of nonconstructive methods of proof in mathematics has a long and dramatic history. In 1888 a young German mathematician David Hilbert presented to his colleagues three short papers on invariant theory. Invariant theory was the highly estimated achievement of Paul Gordan who had produced highly complicated constructive proofs but left several important open problems. The young David Hilbert had solved all these problems and had done much, much more. Paul Gordan was furious. He was not ready to accept the new solutions

* The research was supported by Grant No. 09.1570 from the Latvian Council of Science and by Project 2009/0216/1DP/1.1.2.1.2/09/IPIA/VIA/004 from the European Social Fund.

C.S. Calude et al. (Eds.): UC 2010, LNCS 6079, pp. 11–20, 2010.

because they provided no explicit constructions. Hilbert merely proved that the solutions cannot fail to exist. Gordan refused to accept this as mathematics. He even used the term "theology" and categorically objected to publication of these papers. Nonetheless the papers were published first in *Götingen Nachrichten* and later, in final form, in [15].

In the nineteen-forties the situation, however, changed. In spite of all philosophical battles the nonconstructive methods found their way even to discrete mathematics. This was particularly surprising because here all the objects were finite and it seemed that no kind of distinction between *actual infinity* and *potential infinity* could influence these proofs while most of the discussions between intuitionists and classicists were around these notions. Paul Erdös produced many nice nonconstructive proofs, the first paper of this kind being [7]. Many such proofs are considered in a survey paper by Joel Spancer [24] and a recent monograph by Noga Alon and Joel H. Spencer [2].

R. Karp and R. Lipton have introduced in [16] a notion *Turing machine that takes advice* which is in fact a usage of a nonconstructive help from outside in a process of computation. Later C. Damm and M. Holzer [5] have adapted this notion of advice for finite automata. The adaptation was performed in the most straightforward way (what is quite natural) and later extensively used by T.Yamakami and his coauthors [27,21,26].

Another version of the notion *a finite automaton that takes advice* was introduced in [13] under the name *nonconstructive finite automaton*. These notions are equivalent for large amounts of nonconstructivity (or large amounts of advice) but, for the notion introduced in [5] languages recognizable with polynomial advice are the same languages which are recognizable with a constant advice. The notion of the amount of nonconstructivity in [13] is such that the most interesting results concern the smallest possible amounts of nonconstructivity.

There was a similar situation in the nineteen sixties with space complexity of Turing machines. At first space complexity was considered for one-tape offline Turing machines and it turned out that space complexity is never less than linear. However, it is difficult to prove such lower bounds. Then the seminal paper by R.E.Stearns, J.Hartmanis and P.M.Lewis [25] was published and many-tape Turing machines became a standard tool to study sublinear space complexity.

2 Old Definitions

The essence of nonconstructive methods is as follows. An algorithm is presented in a situation where (seemingly) no algorithm is possible. However, this algorithm has an additional input where a special help is fed in. If this help is correct, the algorithm works correctly. On the other hand, this help on the additional input does not just provide the answer. There still remains much work for the algorithm.

Is this nonconstructivism merely a version of nondeterminism? Not at all. Nondeterministic finite automata (both with 1-way and 2-way inputs) recognize only regular languages while nonconstructive finite automata (as defined in

[13,5]) can recognize some nonregular and even nonrecursive languages. We will see below that this notion is different also from probabilistic finite automata.

Definition 1. *We say that an automaton A recognizes the language L nonconstructively if the automaton A has an input tape where a word x is read and an additional input tape for nonconstructive help y with the following property. For arbitrary natural number n there is a word y such that for all words x whose length does not exceed n the automaton A on the pair (x, y) produces the result 1 if $x \in L$, and A produces the result 0 if $x \notin L$. Technically, the word y can be a tuple of several words and may be placed on separate additional input tapes.*

Definition 2. *We say that an automaton A recognizes the language L nonconstructively with nonconstructivity $d(n)$ if the automaton A has an input tape where a word x is read and an additional input tape for nonconstructive help y with the following property. For arbitrary natural number n there is a word y of the length not exceeding $d(n)$ such that for all words x whose length does not exceed n the automaton A on the pair $(x.y)$ produces the result 1 if $x \in L$, and A produces the result 0 if $x \notin L$. Technically, the word y can be a tuple of several words and may be placed on separate additional input tapes. In this case, $d(n)$ is the upper bound for the total of the lengths of these words.*

The automaton A in these definitions can be a finite automaton, a Turing machine or any other type of automata or machines. In this paper we restrict ourselves by considering only deterministic finite automata with 2-way behavior on each of the tapes.

It turns out that for some languages the nonconstructive help can bring zero information about the input word's being or not being in the language considered. In this paper we try to understand how much help can be done by sending merely a random sequence of bits to a finite automaton. Is it equivalent to the automaton being a probabilistic automaton? (Theorem 1 below shows that it is not.)

Martin-Löf's original definition of a random sequence was in terms of constructive null covers; he defined a sequence to be random if it is not contained in any such cover. Leonid Levin and Claus-Peter Schnorr proved a characterization in terms of Kolmogorov complexity: a sequence is random if there is a uniform bound on the compressibility of its initial segments. An infinite sequence S is Martin-Löf random if and only if there is a constant c such that all of S's finite prefixes are c-incompressible. Schnorr gave a third equivalent definition in terms of martingales (a type of betting strategy). M.Li and P.Vitanyi's book [20] is an excellent introduction to these ideas.

3 Results

C. Dwork and L. Stockmeyer prove in [6] a theorem useful for us:

Theorem A. [6] *Let $L \subseteq \Sigma^*$. Suppose there is an infinite set I of positive integers and, for each $m \in I$, an integer $N(m)$ and sets $W_m = \{w_1, w_2, \cdots, w_{N(m)}\}$, $U_m = \{u_1, u_2, \cdots, u_{N(m)}\}$ and $V_m = \{v_1, v_2, \cdots, v_{N(m)}\}$ of words such that*

1. $| w | \leq m$ *for all* $w \in W_m$*,*

2. for every integer k *there is an* m_k *such that* $N(m) \geq m^k$ *for all* $m \in I$ *with* $m \geq m_k$*, and*

3. for all $1 \leq i, j \leq N(m), u_j w_i v_j$ *iff* $i = j$*.*

Then $L \notin AM(2pfa)$*.*

We use this result to prove

Theorem 1. *[14] (1)The language* $L = \{x2x \mid x \in \{0,1\}^*\}$ *cannot be recognized with a bounded error by a probabilistic 2-way finite automaton.*
(2) The language $L = \{x2x \mid x \in \{0,1\}^*\}$ *can be recognized by a deterministic non-writing 2-tape finite automaton one tape of which contains the input word, and the other tape contains an infinite Martin-Löf random sequence, the automaton is 2-way on every tape, and it stops producing the correct result in a finite number of steps for arbitrary input word.*

Proof. (1) Let m be an arbitrary integer. For arbitrary $i \in \{0, 1, 2, \cdots, 2^m - 1\}$ we define the word $x_i(m)$ as the word number i in the lexicographical ordering of all the binary words of length m. We define the words u_i, w_i, v_i in our usage of Theorem A as $\{\emptyset, x_i(m), 2x_i(m)\}$.

(2) Let the input word be $x(r)2z(s)$ where r and s are the lengths of the corresponding words. At first, the 2-tape automaton finds a fragment $01111\cdots$ which has the length at least r and uses it as a counter to test whether $r = s$. Then the automaton searches for another help-word. If the help-word turns out to be y then the automaton tests whether $x(r) = y$ and whether $z(s) = y$. □

The definition used in the second item of Theorem 1 is our first (but not final) attempt to formalize the main idea of the notion of help from outside bringing zero information about the problem to be solved. Unfortunately, this definition allows something that was not intended to use. Such automata can easily simulate a counter, and 2-way automata with a counter, of course, can recognize nonregular languages. On the other hand, the language L in our Theorem 1 cannot be recognized by a finite automaton with one counter. Hence we try to present a more complicated definition of help from outside bringing zero information to avoid the possibility to simulate a counter.

Definition 3. *A bi-infinite sequence of bits is a sequence* $\{a_i\}$ *where* $i \in (-\infty, \infty)$ *and all* $a_i \in \{0, 1\}$*.*

Definition 4. *We say that a bi-infinite sequence of bits is Martin-Löf random if for arbitrary* $i \in (-\infty, \infty)$ *the sequence* $\{b_n\}$ *where* $b_n = a_{i+n}$ *for all* $i \in \mathbb{N}$ *is Martin-Löf random, and the sequence* $\{c_n\}$ *where* $c_n = a_{i-n}$ *for all* $i \in \mathbb{N}$ *is Martin-Löf random.*

Definition 5. *A deterministic finite automaton with intuition is a deterministic non-writing 2-tape finite automaton one tape of which contains the input* *word, and the other tape contains a bi-infinite Martin-Löf random sequence, the automaton is 2-way on every tape, and it stops producing a the correct result in a finite number of steps for arbitrary input word. Additionally it is demanded that the head of the automaton never goes beyond the markers showing the beginning and the end of the input word.*

Definition 6. *A deterministic finite-state transducer with intuition is a deterministic 3-tape finite automaton one non-writing tape of which contains the input word, and the other non-writing tape contains a bi-infinite Martin-Löf random sequence, the automaton is 2-way on these tapes, and a third writing non-reading one-way tape for output. The automaton produces a the correct result in a finite number of steps for all input words. Additionally it is demanded that the head of the automaton never goes beyond the markers showing the beginning and the end of the input word.*

Nondeterministic, probabilistic, alternating, etc. automata with intuition differ from deterministic ones only in the nature of the automata but not in usage of tapes or Martin-Löf random sequences.

Definition 7. *We say that a language L is recognizable by a deterministic finite automaton A with intuition if A for arbitrary bi-infinite Martin-Löf random sequence accepts every input word $x \in L$ and rejects every input word $x \notin L$.*

Definition 8. *We say that a language L is enumerable by a deterministic finite automaton A with intuition if A for arbitrary bi-infinite Martin-Löf random sequence accepts every input word $x \in L$ and does not accept any input word $x \notin L$.*

Definition 9. *A deterministic finite automaton with intuition on unbounded input is a deterministic non-writing 2-tape finite automaton one tape of which contains the input word, and the other tape contains a bi-infinite Martin-Löf random sequence, the automaton is 2-way on every tape, and it produces a the correct result in a finite number of steps for arbitrary input word. It is not demanded that the head of the automaton always remains between the markers showing the beginning and the end of the input word.*

Recognition and enumeration of languages by deterministic finite automata with intuition is not particularly interesting because of the following two theorems.

Theorem 2. *A language L is enumerable by a deterministic finite automaton with intuition on unbounded input if and only if it is recursively enumerable.*

Proof. J.Bārzdiņš [4] proved that arbitrary one-tape deterministic Turing machine can be simulated by a 2-way finite deterministic automaton with 3 counters directly and by a 2-way finite deterministic automaton with 2 counters using a simple coding of the input word. (Later essentially the same result was re-discovered by other authors.) Hence there exists a a 2-way finite deterministic automaton with 3 counters accepting every word in L and only words in L.

Let x be an arbitrary word in L. To describe the processing of x by the 3-couter automaton we denote the content of the counter i ($i \in \{1,2,3\}$) at the moment t by $d(i,t)$. The word

$$00000101^{d(1,0)}0101^{d(2,0)}0101^{d(3,0)}000101^{d(1,1)}0101^{d(2,1)}0101^{d(3,1)}00\cdots$$

$$\cdots 00101^{d(1,s)}0101^{d(2,s)}0101^{d(3,s)0000}$$

where s is the halting moment, is a complete description of the processing of x by the automaton.

Our automaton with intuition tries to find a fragment of the bi-infinite Martin-Löf random sequence on the help-tape such that:

1. it starts and ends by 0000,
2. the initial fragment
$$0101^{d(1,0)}0101^{d(2,0)}0101^{d(3,0)}00$$
is exactly 0000010010010, (i.e., the all 3 counters are empty,
3. for arbitrary t the fragment
$$0101^{d(1,t)}0101^{d(2,t)}0101^{d(3,t)}0101^{d(1,t+1)}0101^{d(2,t+1)}0101^{d(3,t+1)}$$
coresponds to a legal instruction of the automaton with the counters.

Since the bi-infinite sequence is Martin-Löf random, such a fragment definitely exists in the sequence infinitely many times. The correctness of the fragment can be tested using the 3 auxiliary constructions below.

Construction 1. Assume that $w_k \in \{0,1\}^*$ and $w_m \in \{0,1\}^*$ are two subwords of the input word x such that:

1. they are immediately preceded and immediately followed by symbols other than $\{0,1\}$,
2. a deterministic finite 1-tape 2-way automaton has no difficulty to move from w_k to w_m and back, clearly identifying these subwords,

Then there is a deterministic finite automaton with intuition recognizing whether or not $w_k = w_m$.

Proof. As in Theorem 1. □

Construction 2. Assume that 1^k and 1^m are two subwords of the help-word y such that:

1. they are immediately preceded and immediately followed by symbols other than $\{0,1\}$,

2. a deterministic finite 1-tape 2-way automaton has no difficulty to move from w_k to w_m and back, clearly identifying these subwords,

3. both k and m are integers not exceeding the length of the input word.

Then there is a deterministic finite automaton with intuition recognizing whether or not $k = m$.

Proof. Similar the proof of Construction 1.

Construction 3. Assume that $1^{k_1}, 1^{k_2}, \cdots, 1^{k_s}$ and $1^{m_1}, 1^{m_2}, \cdots, 1^{m_t}$ are subwords of the help-word y such that:

1. they are immediately preceded and immediately followed by symbols other than 1,
2. a deterministic finite 1-tape 2-way automaton has no difficulty to move from one subword to another and back, clearly identifying these subwords,

3. both $k_1 + k_2 + \cdots + k_s$ and $m_1 + m_2 + \cdots + m_t$ are integers not exceeding the length of the input word.

Then there is a deterministic finite automaton with intuition recognizing whether or not $k_1 + k_2 + \cdots + k_s = m_1 + m_2 + \cdots + m_t$.

Proof. Similar the proof of Construction 2. □

Corollary of Theorem 2. *A language L is recognizable by a deterministic finite automaton with intuition on unbounded input if and only if it is recursive.*

Theorem 2 and its corollary show that the standard definition of the automaton with intuition should avoid the possibility to use the input tape outside the markers. However, even our standard definition allows recognition and enumeration of nontrivial languages. The proof of Theorem 1 can be easily modified to prove

Theorem 3. *[14]*

1. *The language $L = \{x2x \mid x \in \{0,1\}^*\}$ cannot be recognized with a bounded error by a probabilistic 2-way finite automaton,*

2. *The language $L = \{x2x \mid x \in \{0,1\}^*\}$ can be recognized by a deterministic finite automaton with intuition.*

Theorem 4. *[14] The unary language PERFECT SQUARES* $= \{1^n \mid (\exists m)(n = m^2)\}$ *can be recognized by a deterministic finite automaton with intuition.*

Proof. It is well-known that

$$1 + 3 + 5 + \cdots + (2n - 1) = n^2.$$

The deterministic automaton with intuition searches for a help-word (being a fragment of the given bi-infinite Martin-Löf sequence) of a help-word

$$00101110111110 \cdots 01^{2n-1}00.$$

At first, the input word is used as a counter to test whether each substring of 1's is exactly 2 symbols longer than the preceding one. Then the help-word is used to test whether the length of the input word coinsides with the number of 1's in the help-word. □

The proof of this theorem can be modified to prove that square roots can be extracted by deterministic finite-state transducers with intuition.

Theorem 5. *The relation SQUARE ROOTS* $= \{(1^n, 1^m) \mid (n = m^2)\}$ *can be computed by a deterministic finite-state transducer with intuition.*

Proof. The output can easily be obtained from the help-word in the preceding proof. □

Theorem 6. *The relation CUBE ROOTS* $= \{(1^n, 1^m) \mid (n = m^3)\}$ *can be computed by a deterministic finite-state transducer with intuition.*

Proof. In a similar manner the formula

$$1 + 3(n-1) + 3(n-1)^2 = n^3$$

suggests a help-word

$$000[1]00[101110111]00[1011111101111111111111]00\cdots00[101^{n-1}01^{(n-1)^2}]000$$

where symbols [,] are invisible. At first, the input word is used as a counter to test whether the help-word is correct but not whether its length is sufficient. Then the help-word is used to test whether the length of the input word coinsides with the number of 1's in the help-word. □

Theorem 7. *The relation*

$$FACTORISATION = \{(1^n, 1^m) \mid (m \textit{ divides } n \wedge m \neq 1) \vee (m = 1 \textit{ if } m \textit{ is prime})\}$$

can be computed by a deterministic finite-state transducer with intuition.

We define a relation UNARY 3-SATISFIABILITY as follows. The term $term_1 = x_k$ is coded as $[term_1]$ being 21^k, the term $term_2 = \neg x_k$ is coded as $[term_2]$ being 31^k, the subformula f being $(term_1 \vee term_2 \vee term_3)$ is coded as $[f]$ being $[term_1] \vee [term_2] \vee [term_3]$. The CNF being $f_1 \wedge f_2 \wedge \cdots \wedge f_m$ is coded as $[f_1] \wedge [f_2] \wedge \cdots \wedge [f_m]$. The string of the values $x_1 = a_1, x_2 = a_2, \cdots, x_n = a_n$ is coded as $a_1 a_2 \cdots a_n$. The relation UNARY 3-SATISFIABILITY consists of all the pairs $(CNF, a_1 a_2 \cdots a_n)$ such that the given CNF with these values of the arguments takes the value TRUE.

Theorem 8. *The relation UNARY 3-SATISFIABILITY can be computed by a deterministic finite-state transducer with intuition.*

References

1. Ablayev, F.M., Freivalds, R.: Why Sometimes Probabilistic Algorithms Can Be More Effective. LNCS, vol. 233, pp. 1–14. Springer, Heidelberg (1986)
2. Alon, N., Spencer, J.H.: The Probabilistic Method. John Wiley & Sons, Chichester (2000)
3. Bach, E., Shallit, J.: Algorithmic Number Theory, vol. 1. MIT Press, Cambridge (1996)
4. Bārzdiņš, J. (Barzdin, J.M.): On a Class of Turing Machines (Minsky Machines). Algebra i Logika 3(1) (1963) (Russian); Review in The Journal of Symbolic Logic 32(4), 523–524 (1967)
5. Damm, C., Holzer, M.: Automata that take advice. In: Hájek, P., Wiedermann, J. (eds.) MFCS 1995. LNCS, vol. 969, pp. 565–613. Springer, Heidelberg (1995)
6. Dwork, C., Stockmeyer, L.: Finite state verifiers I: the power of interaction. Journal of the Association for Computing Machinery 39(4), 800–828 (1992)
7. Erdös, P.: Some remarks on the theory of graphs. Bulletin of the American Mathematical Society 53(4), 292–294 (1947)
8. Fagin, R.: Generalized First-Order Spectra and Polynomial-Time Recognizable Sets. In: Karp, R. (ed.) SIAM-AMS Proceedings of Complexity of Computation, vol. 7, pp. 27–41 (1974)
9. Freivalds, R. (Freivald, R.V.): Recognition of languages with high probability on different classes of automata. Dolady Akademii Nauk SSSR 239(1), 60–62 (1978) (Russian)
10. Freivalds, R.: Projections of Languages Recognizable by Probabilistic and Alternating Finite Multitape Automata. Information Processing Letters 13(4/5), 195–198 (1981)
11. Freivalds, R.: Complexity of Probabilistic Versus Deterministic Automata. In: Barzdins, J., Bjorner, D. (eds.) Baltic Computer Science. LNCS, vol. 502, pp. 565–613. Springer, Heidelberg (1991)
12. Freivalds, R.: Non-Constructive Methods for Finite Probabilistic Automata. International Journal of Foundations of Computer Science 19(3), 565–580 (2008)
13. Freivalds, R.: Amount of nonconstructivity in finite automata. In: Maneth, S. (ed.) CIAA 2009. LNCS, vol. 5642, pp. 227–236. Springer, Heidelberg (2009)
14. Freivalds, R.: Multiple usage of random bits by finite automata. Unpublished manuscript (2010)
15. Hilbert, D.: Uber die Theorie der algebraischen Formen. Mathematische Annalen 36, 473–534 (1890)
16. Karp, R.M., Lipton, R.: Turing machines that take advice. L' Enseignement Mathematique 28, 191–209 (1982)
17. Kolmogorov, A.N.: Three approaches to the quantitative definition of information. Problems in Information Transmission 1, 1–7 (1965)
18. Levin, L.A.: On the notion of a random sequence. Soviet Mathematics Doklady 14, 1413–1416 (1973)
19. Martin-Löf, P.: The definition of random sequences. Information and Control 9(6), 602–619 (1966)
20. Li, M., Vitányi, P.M.B.: An Introduction to Kolmogorov Complexity and its Applications, 2nd edn. Springer, Heidelberg (1997)
21. Nishimura, H., Yamakami, T.: Polynomial time quantum computation with advice. Information Processing Letters 90(4), 195–204 (2004)

22. Schnorr, C.-P.: A unified approach to the definition of random sequences. Mathematical Systems Theory 5(3), 246–258 (1971)
23. Schnorr, C.-P.: Process Complexity and Effective Random Tests. Journal of Computer and System Sciences 7(4), 376–388 (1973)
24. Spencer, J.: Nonconstructive methods in discrete mathematics. In: Rota, G.-C. (ed.) Studies in Mathematics, MAA, vol. 17, pp. 142–178 (1978)
25. Stearns, R.E., Hartmanis, J., Lewis II, P.M.: Hierarchies of memory limited computations. In: Proceedings of FOCS, pp. 179–190 (1965)
26. Tadaki, K., Yamakami, T., Lin, J.C.H.: Theory of One Tape Linear Time Turing Machines. In: Van Emde Boas, P., Pokorný, J., Bieliková, M., Štuller, J. (eds.) SOFSEM 2004. LNCS, vol. 2932, pp. 335–348. Springer, Heidelberg (2004)
27. Yamakami, T.: Swapping lemmas for regular and context-free languages with advice. The Computing Research Repository (CoRR), CoRR abs/0808.4122 (2008)