Randomization and nondeterminsm are incomparable for ordered read-once branching programs

Farid Ablayev[*]

Department of Theoretical Cybernetics
Kazan University
Kazan 420008, Russia
ablayev@ksu.ru

Abstract

In [3] we exhibited a simple boolean functions f_n in n variables such that:

1) f_n can be computed by polynomial size randomized ordered read-once branching program with one sided small error;

2) any nondeterministic ordered read-once branching program that computes f_n has exponential size.

In this paper we present a simple boolean function g_n in n variables such that:

1) g_n can be computed by polynomial size nondeterministic ordered read-once branching program;

2) any two-sided error randomized ordered read-once branching program that computes f_n has exponential size.

These mean that BPP and NP are incomparable in the context of ordered read-once branching program.

1 Preliminaries

Branching programs is well known model of computation for discrete functions [14]. Many types of restricted branching programs have been investigated as

[*] Work done in part while visiting Steklov Mathematical Institute in Moscow. The research supported by Russia Fund for Basic Research 96-01-01962.

important theoretical model of computations [9]. Ordered read-once branching program or ordered binary decision diagrams (OBDD) [4, 15] also important for practical computer science. They are used in circuits verifications. But many important functions cannot be computed by determinsitc read-once branching programs of polynomial size [4, 13, 8].

In [2] we introduced the model of randomized branching programs and showed that randomized ordered read-once branching programs can be more effective than determinstic ones. In [3] we defined exclusive boolean function f_n in n variables which can be computed by polynomial size randomized ordered read-once branching program, but any nondeterminstic ordered read-once branching program needs exponetial size to compute f_n. Martin Sauerhoff [10] considered function from theorem 3 [6]. He proved that this function needs (also as in the deterministic case) exponetial size randomized read-once branching programs for one-sided error. In this paper we presented exclusive function g_n which is "simple" for nondeterminstic ordered read-once branching programs, but is "hard" for randomized read-once branching programs with two-sided error of computation.

Together with the result from [3] this proves that complexity classes BPP and NP are incomparable in the context of ordered read-once branching programs.

Note that the results of the paper for ordered read-once branching programs are true for a more common model — weak-ordered branching program that we define in the paper. Informaly speaking weak-ordered property for branching program P means existence of partition of its set $\{x_1, x_2, \ldots, x_n\}$ of variables into two parts X_1 and X_2, $X_1 \bigcap X_2 \neq \emptyset$, such that for any computation path of P the following is true. If a variable from X_2 is tested then no variable from X_1 can be tested in the rest part of this path.

A *deterministic* branching program P for computing a function $g : \{0,1\}^n \rightarrow \{0,1\}$ is a directed acyclic multi-graph with a distinguished source node s and a distinguished sink node t. The out degree of of each non-sink node is exactly 2 and the two outgoing edges are labeled by $x_i = 0$ and $x_i = 1$ for variable x_i associated with the node. Call such node an x_i-node. The label "$x_i = \delta$" indicates that only inputs satisfying $x_i = \delta$ may follow this edge in the computation. The branching program P computes function g in the obvious way: for each $\sigma \in \{0,1\}^n$ we let $f(\sigma) = 1$ iff there is a directed $s - t$ path starting in the source s and leading to to the accepting node t such that all labels $x_i = \sigma_i$ along this path are consistent with $\sigma = \sigma_1, \sigma_2, \ldots, \sigma_n$.

The branching program becomes *nondeterministic* [5] if we allow "guessing nodes" that is nodes with two outgoing edges being unlabeled. Unlabeled edges allow all inputs to produced. A nondeterministic branching program P computes a function g, in the obvious way; that is, $g(\sigma) = 1$ iff there exists (at least one) computation on σ starting in the source node s and leading to the accepting node t.

Define a *randomized* branching program [2] as a one which has in addition to its standard inputs specially designated inputs called "random inputs". When values of these "random inputs" are chosen from the uniform distribution, the output of the branching program is a random variable.

Say that a randomized branching program (a, b)-computes a boolean function f if it outputs 1 with probability at most a for input σ such that $f(\sigma) = 0$ and outputs 1 with probability at least b for inputs σ such that $f(\sigma) = 1$.

As usual for a branching program P (deterministic or random), we define size(P) (complexity of the branching program P) as the number of internal nodes in P. Define, following [5], the size(P) of the nondeterminstic branching program P as the number of internal nodes in P minus the number of guessing nodes.

Read-once branching programs is branching program in which for each path each variable is tested no more than once. An ordered read-once branching program is a read-once branching program which respects a fixed ordering π of the variables, i.e. if an edge leads from an x_i-node to an x_j-node, the condition $\pi(i) < \pi(j)$ has to be fulfilled.

2 Results

We specify a boolean function f_n of $n = 4l$ variables as follows. For a sequence $\sigma \in \{0,1\}^{4l}$ call odd bits a "type" bits and even bits a "value" bits. Say that even bit $\sigma_i \in \sigma$, $i \in \{2, 4, \ldots, 4l\}$, has type 0 (1) if corresponding odd bit σ_{i-1} is 0 (1). For a sequence $\sigma \in \{0,1\}^{4l}$ denote σ^0 (σ^1) subsequence of σ that consists of all even bits of type 0 (1).

For every $\sigma \in \{0,1\}^n$ boolean function $f_n : \{0,1\}^n \to \{0,1\}$ is defined as $f_n(\sigma) = 1$ iff $\sigma^0 = \sigma^1$.

Definition 1 *Call branching program a π-weak-ordered branching program if its respects a partition π of variables $\{x_1, x_2, \ldots, x_n\}$ into two parts X_1 and X_2 such that if an edge leads from an x_i-node to an x_j-node, where $x_i \in X_t$ and $x_j \in X_m$, then the condition $t \leq m$ has to be fulfilled.*

Call branching program P an weak-ordered if it is π-weak-ordered for some partition π of the set of variables of P into two sets.

Clearly that ordered read-once branching program is also weak-ordered. We proved the following result in [3] (we use here a restrictive variant of this result).

Theorem 1 *For the function f_n the following is true:*

1. f_n can be $(\varepsilon(n), 1)$-computed by randomized ordered read-once branching program of the size

$$O\left(\frac{n^6}{\varepsilon^3(n)} \log^2 \frac{n}{\varepsilon(n)}\right).$$

2. Any nondeterministic ordered read-once branching program that computes function f_n has the size no less than $2^{n/4-1}$.

Now define function g_n which is "hard" for randomized computation but is "simple" for nondeterminstic computation for our model of branching program. This boolean function presented in [11]. Let n be an integer and let $p[n]$ be the smallest prime greater or equal to n. Then, for every integer s, let $\omega_n(s)$ be defined as follows. Let j be the unique integer satisfying $j = s \bmod p[n]$ and $1 \leq j \leq p[n]$. Then, $\omega_n(s) = j$, if $1 \leq j \leq n$, and $\omega_n(s) = 1$ otherwise.

For every n, the boolean function $g_n : \{0,1\}^n \to \{0,1\}$ is defined as $g_n(\sigma) = \sigma_j$, where $j = \omega_n(\sum_{i=1}^n i\sigma_i)$.

We will use the following notations in the rest part of the paper. Let $h : \{0,1\}^n \to \{0,1\}$ be a boolean function. Consider a partition π of variables $\{x_1, x_2, \ldots, x_n\}$ into two parts $X_1 = \{x_i : i \in I\}$ and $X_2 = \{x_j : j \in J\}$, where $I \subset \{1, 2, \ldots, n\}$, $|I| = l$ and $J = \{1, 2, \ldots, n\} \backslash I\}$, $|J| = t$.

Denote L, R sets of binary sequences of length l and t with indexes from I and J respectively. For $u \in L$ and $w \in R$ let (u, w) mean the sequence σ from $\{0,1\}^n$ in wich bits with indexes from I respectively J have the same values as in u respectively w. We will also use the notation $h(u, w)$ instead of $h(\sigma)$ where it will be convenient.

Consider one-way randomized communication computation. We use the following standard model of one-way randomized communication computation for function h. Two players A and B receive respectively $u \in L$ and $w \in R$. In the randomized one-way model, A sends the messages $\beta_1, \beta_2, ..., \beta_d$ with probabilities $p_1, p_2, ..., p_d$ respectively ($\sum_{i=1}^d p_i = 1$). B, on the receipt of β_i, outputs 1 with probability q_i and 0 with probability $1 - q_i$. The probability distribution on the set of messages sent by A is entirely determined by the input at A alone, and is not influenced by the input at B. Similarly, the probabilities q_i at B depend only on its input and the message β_i received.

In the computation $T_\phi(u, w)$, the probability of outputting the bit $b = 1$ is $\sum_{i=1}^d p_i(u)q_i(w)$ and the bit $b = 0$ is $1 - \sum_{i=1}^d p_i(u)q_i(w)$.

Let $p = \frac{1}{2} + \varepsilon$ for $0 \leq \varepsilon \leq 1/2$. Say that the probabilistic protocol ϕ p-computes a function h if for every input $\sigma = (u, w)$ it holds that $h(\sigma) = b$ iff the probability of outputting the bit b in the computation $T_\phi(u, w)$ is no less than p.

Let a set $U \subseteq \{0,1\}^n$ be such that $U = L \times R$. The randomized communication complexity $C(\phi)$ of the probabilistic protocol ϕ on the inputs from U is $\lceil \log |M(\phi)| \rceil$, where $M(\phi)$ is the set of messages used by ϕ during computations on inputs from U. For $p \in [1/2, 1]$ the randomized communication complexity $PC^U_{p,\pi}(h)$ of a boolean function h is

$$\min\{C(\phi) : \text{ protocol } \phi \text{ } p\text{-computes } h \text{ for the partion } \pi \text{ of inputs from } U\}.$$

The proof of following lemma is based on simulation technique of weak-ordered branching program by communication protocol and is similar to simulation technique from [1] (lemma 6.1).

Lemma 1 *Let $\varepsilon \in [0,1/2]$, $p = 1/2+\varepsilon$. Let randomized π-weak-ordered branching program P $(1-p,p)$-computes function $h : \{0,1\}^n \to \{0,1\}$. Let $U \subseteq \{0,1\}^n$ be such that $U = L \times R$, where L and R are defined in according to partition π of inputs. Then*

$$size(P) \geq 2^{PC^U_{p,\pi}(h)-1}.$$

Proof. Describe the following communication protocol Φ, which p-computes function h for the partion π of inputs.

Let $\sigma \in U$ be a valuation of x, $\sigma = (u,w)$, $u \in L$, $w \in R$. Players A and B receive respectively u and w in according to partition π of inputs. Let $v_1,\ldots,v_d$ be all internal nodes of P that are reachable during paths of computation on the part u of input σ with non zero probabilities $p_1(u),\ldots,p_d(u)$.

During the computation on the input u, player A sends node v_i with probability $p_i(u)$ to player B. Player B on obtaining message v_i from A starts its computation (simulation of the branching program P) from the node v_i on the part w of the input σ.

From the definition of the protocol Φ results the statement of the lemma. ▮

We use the lower bound for probabilistic one-way complexity from [1] in the proof of the theorem 3 below. Recall notations and the statement we need from [1] in the convinient for us form.

For $U = L \times R$ with a boolean function h we associate a $|L| \times |R|$ communication matrix CM whose (u,w)-th entry, $CM[u,w]$ is $h(u,w)$ for all $(u,w) \in L \times R$. As it is mentioned in [16] the one-way deterministic communication complexity $DC^U_\pi(h)$ for partition π of inputs from U of a boolean function h is easily seen to be $\lceil \log(nrow(CM)) \rceil$, where $nrow(CM)$ is the number of distinct rows of communication matrix CM of the function h.

Consider w.l.g. the case when all rows of CM are different, $nrow(CM) = |L|$.

Choose a $Y \subseteq R$ such that for an arbitrary two words $u, u' \in L$ there exists a word $y \in Y$ such that $h(u,y) \neq h(u',y)$. *The set Y is called the control set for the matrix CM.*

Denote

$$ts(CM) = \min\{|Y| : Y \text{ } is \text{ } a \text{ } control \text{ } set \text{ } for \text{ } CM\}.$$

It is evident that $\lceil \log nrow(CM) \rceil \leq ts(CM) \leq nrow(CM)$.

For number $p \in [1/2,1]$, define $pcc^U_p(h) = \frac{ts(CM)}{\log nrow(CM)} H(p)$, where $H(p) = -p\log p - (1-p)\log(1-p)$ is the Shannon entropy. Call $pcc^U_p(h)$ the p-probabilistic communication characteristic of the function h.

Theorem 2 *[1] Let $\varepsilon \in [0,1/2]$, $p = 1/2+\varepsilon$. Let $U \subseteq \{0,1\}^n$ be such that $U = L \times R$, where L and R are defined in according to partition π of inputs of function $h : \{0,1\}^n \to \{0,1\}$. Then*

$$PC^U_{p,\pi}(h) \geq DC^U_\pi(h)(1 - pcc^U_p(h)) - 1.$$

In the proof of the theorem 3 below we use the following result from number theory (see [7] and [12] for additional citation).

For every natural number n let $p(n)$ be the smalest prime greater or equal than n. Consider $Z_{p(n)}$ the field of the residue classes modulo p.

Lemma 2 *For every n large enough, the following is true. If $A \subseteq Z_{p(n)}$ and $|A| \geq 3\sqrt{n}$, then, for every $t \in Z_{p(n)}$, there is a subset $B \subseteq A$ such that the sum of the elements of B is equal to t.*

Theorem 3 *Let $\varepsilon \in [0, 1/2]$, $p = 1/2 + \varepsilon$. Then for arbitrary $\delta > 0$ for every n large enough it holds that any randomized ordered read-once branching program that $(1-p, p)$-computes function g_n has the size no less than*

$$1/4 \left(\frac{2^{n - \lceil 3\sqrt{n} \rceil}}{n} \right)^{1-(1+\delta)H(p)}.$$

Proof. Let P be a randomized ordered read-once branching program with an ordering τ of variables which computes function g_n. For ordering $\tau = \{i_1, i_2, \ldots, i_n\}$ consider the partition π of variables x of g_n into two parts $X_1 = \{x_{i_1}, \ldots, x_{i_l}\}$ and $X_2 = \{x_{i_{l+1}}, \ldots, x_{i_n}\}$, where $l = n - \lceil 3\sqrt{n} \rceil$. Denote $t = \lceil 3\sqrt{n} \rceil$.

Describe below a subset $U \subset \{0,1\}^n$ in the form $U = L \times R$ where $|L| = l$, $|R| = t$.

Denote by I and J sets of indexes of variables from sets X_1 and X_2 respectively. For $s \in \{1, \ldots, n\}$ denote L_s a subset of binary sequences of length l with indexes from I such that $L_s = \{u : \omega_n(\sum_{i \in I} iu_i) = s\}$. Denote L a maximum among sets $L_1, \ldots, L_n$.

$$|L| = \max_{s \in \{1, \ldots, n\}} \{|L_s|\}.$$

Clearly that

$$|L| \geq \frac{2^{n - \lceil 3\sqrt{n} \rceil}}{n}.$$

Let $L = L_s$. Then denote $R = \{w : \omega_n(\sum_{j \in J} jw_j + s) = k, k \in I\}$. From the definition of R we have the following properties:

1) $|R| = l$;

2) for arbitrary u and u' from L there exists $w \in R$ such that $g_n(u, w) \neq g_n(u', w)$.

We will prove the second property (the first one is evident). Let $i \in I$ be an index such that i-th bits in sequences u and u' are different, $u_i \neq u'_i$. From the lemma 2 it follows that for every n large enough, for our number s and the number i there exists a sequence $w \in R$ such that $s + \sum_{j \in J} jw_j = i \bmod p(n)$. Then from the definition of g_n it follows that $g_n(u, w) \neq g_n(u', w)$.

Now define set U as $U = L \times R$. From the above it follows that for the set U $|L| \times |R|$ communication matrix CM of g_n has the following properties:

1) $nrow(CM) = |L|$;

2) the set R is the control set for CM.

This means that $DC^U(g_n) = \log |L|$ and that for p-probabilistic communication characteristic of $pcc_p^U(g_n)$ of function g_n it is true that

$$pcc_p^U(g_n) = (l/\log |L|)H(p) \leq ((n - \lceil 3\sqrt{n} \rceil)/(n - \lceil 3\sqrt{n} \rceil - \log n))H(p).$$

From this it follows that for arbitrary $\delta > 0$ for every n large enough it holds that

$$pcc_p^U(g_n) \leq (1 + \delta)H(n).$$

From the above property and the theorem 2 it follows that for every n large enough the following is true

$$PC_p^U(g_n) \geq (n - \lceil 3\sqrt{n} \rceil - \log n)(1 - (1 + \delta)H(p)) - 1.$$

From this and the lemma 1 the lower bound for $size(P)$ results. ∎

Note that in the proof of the theorem 3 from the property of P that it is ordered read-once we use only the following fact. Set x of variables of P can be partition into two parts X_1 and X_2 such that $|X_1| = n - \lceil 3\sqrt{n} \rceil$ and $|X_2| = \lceil 3\sqrt{n} \rceil$. The cardinality of X_2 is essential for application of lemma 2. This means that the following statement is true.

Theorem 4 *Let $\varepsilon \in [0, 1/2]$, $p = 1/2 + \varepsilon$. Let P be a randomized π-weak-ordered branching program that $(1 - p, p)$-computes function g_n. Let π be a partition of x in two two parts X_1, X_2 such that $|X_2| = t \geq \lceil 3\sqrt{n} \rceil$ and $|X_1| = l = n - t$. Then for arbitrary $\delta > 0$ for every n large enough it holds that*

$$size(P) \geq 1/4 \left(\frac{2^l}{n} \right)^{1-(1+\delta)H(p)}.$$

Theorem 5 *There is polynomial size nondeterministic ordered read-once branching program that computes function g_n.*

Proof. The proof is simple. For arbitrary input σ nondeterministic ordered read-once branching program P that computes function g_n works as follows. On the first (nondeterminstic) phase P nondeterministicaly selects number $s \in \{1, \ldots, n\}$. Then on the second (deterministic) phase P reads inputs in the order $x_1, \ldots, x_n$. During computation path on input σ P 1) counts number $a = \omega_n(\sum_{i=1}^n i\sigma_i)$ and 2) store s-ths bit σ_s. If $a = s$ then P ouputs bit σ_s of the input σ else P outputs 0. Clearly, that P has polynomial size. ▮

Acknowledgments. I would like to thank Sasha Razborov for his invitation to me to spend my sabbatical semester in Steklov Mathematical Institute and for a number of valuable discussions on the subject of the paper, to participants of Complexity seminar in Steklov Mathematical Institute and Moscow University for listening to results presented in this paper.

References

[1] F.Ablayev, Lower bounds for one-way probabilistic communication complexity and their application to space complexity, *Theoretical Computer Science*, 157, (1996), 139-159.

[2] F. Ablayev and M. Karpinski, On the power of randomized branching programs, *in Proceedings of the ICALP'96, Lecture Notes in Computer Science, Springer-Verlag*, 1099, (1996), 348-356.

[3] F. Ablayev and M. Karpinski, On the power of randomized branching programs, *manuscript* (generalization of ICALP'96 paper results for the case of pure boolean function), available at `http://www.ksu.ru/`˜ablayev .

[4] R. Bryant, Symbolic boolean manipulation with ordered binary decision diagrams, *ACM Computing Surveys*, 24, No. 3, (1992), 293-318.

[5] A. Borodin, A. Razborov, and R. Smolensky, On lower bounds for read-k-times branching programs, *Computational Complexity*, 3, (1993), 1-18.

[6] Y. Breitbart, H.Hunt III, and D. Rosenkratz, On the size of binary decision diagrams representing Boolean functions, *Theoretical Computer Science*, 145, (1995), 45-69.

[7] J. Dias da Silva and Y. Hamidoune, Cyclic spaces for Grassmann derivatives and additive theory, *Bull. London Math. Soc.*, 26, (1994), 140-146.

[8] S. Ponsio, A lower bound for integer multiplication with read-once branching programs, *Proceedings of the 27-th STOC*, (1995), 130-139.

[9] A. Razborov, Lower bounds for deterministic and nondeterministic branching programs, *in Proceedings of the FCT'91, Lecture Notes in Computer Science, Springer-Verlag*, 529, (1991), 47–60.

[10] M. Sauerhoff, Lower bounds for the RP-OBDD-Size, *manuscript*, personal communication.

[11] P. Savicky, S. Zak, A large lower bound for 1-branching programs, *Electronic Colloquium on Computational Complexity*, Revision 01 of TR96-036, (1996), available at `http://www.eccc.uni-trier.de/eccc/` .

[12] P. Savicky, S. Zak, A hierarchy for $(1, +k)$-branching programs with respect to k, *Electronic Colloquium on Computational Complexity*, TR96-050, (1996), available at `http://www.eccc.uni-trier.de/eccc/` .

[13] J. Simon and M. Szegedy, A new lower bound theorem for read-only-once branching programs and its applications, *Advances in Computational Complexity Theory*, ed. Jin-Yi Cai, DIMACS Series, 13, AMS (1993), 183-193.

[14] I. Wegener, *The complexity of Boolean functions.* Wiley-Teubner Series in Comp. Sci., New York – Stuttgart, 1987.

[15] I. Wegener, Efficient data structures for boolean functions, *Discrete Mathematics*, 136, (1994), 347-372.

[16] A. C. Yao, Some Complexity Questions Related to Distributive Computing, *in Proc. of the 11th Annual ACM Symposium on the Theory of Computing*, (1979), 209-213.