Electronic Colloquium on Computational Complexity, Report No. 44 (1999)

On Complexity of Regular $(1, +k)$-Branching Programs

Farid Ablayev[*]

Abstract

A *regular* $(1, +k)$*-branching program* ($(1, +k)$-ReBP) is an ordinary branching program with the following restrictions: (i) along every consistent path at most k variables are tested more than once, (ii) for each node v on all paths from the source to v the same set $X(v) \subseteq X$ of variables is tested, and (iii) on each path from the source to a sink all variables X are tested.

We show that polynomial size $(1, +1)$-ReBP-s are more powerful than polynomial size read-once branching programs and that polynomial size $(1, +(k+1))$-ReBP-s are more powerful than polynomial size $(1, +k)$-ReBP-s.

We prove lower bound $2^{(n-k)/2-k\log(n^2/k)}/2\sqrt{n}$ for $k = o(n^2)$ on the size of any nondeterministic $(1, +k)$-ReBP computing permutation function $PERM_{n^2}$ on n^2 arguments. The proof is based on combination of decomposing of $(1, +k)$-ReBP with communication complexity technique.

Keywords: Branching Programs, OBDD, Circuits complexity, Communication complexity, Lower bounds

1 Preliminaries and definitions

In theory branching programs (for short BP-s) are useful for investigation the amount of space necessary to compute various functions (see survey [13]). Developments in the field of digital design and verification have led to the restricted forms of branching programs. A most common model used for verifying circuits is a polynomial size *ordered read-once branching program* also called an *ordered binary decision diagram* (OBDD).

BP models that requires some kind of ordering of variables testings such as oblivious and OBDD based models are computation models for which communication complexity methods are very productive, see [1, 3, 10, 17]. It is appeared in last decade that communication complexity methods are very productive tool for investigation of power of different models of computations. See books [7, 11] for more information on the subject. Borodin Razborov and Smolensky developed combinatorial method for proving lower bounds for syntactic read-k-times nondeterministic BP-s [2]. Their method can be viewed as a generalization of communication complexity technique. As it is mentioned in [2] in a special cases their method turns to a pure communication complexity technique. The possibility of applying of pure communication complexity methods for nonsyntactic branching programs without ordering of variables testing was open. See, for example, [12] for discussion.

[*] Dept. of Theoretical Cybernetics, Kazan State University. Email: ablayev@ksu.ru. The research supported partially by Russia Fund for Basic Research 99-01-00163

ISSN 1433-8092

In the paper we spread communication complexity methods for proving lower bounds of complexity for branching programs without conditions of ordered testing of variables. Namely we develop communication complexity method for proving lower bounds of complexity for ordinary read-once BP-s and their generalizations $(1,+k)$-ReBP-s. Using our method we prove lower bound $2^{(n-k)/2-k\log(n^2/k)}/2\sqrt{n}$ for $k=o(n^2)$ on the size of any nondeterministic $(1,+k)$-ReBP computing known permutation function $PERM_{n^2}$ on n^2 arguments.

Next we show that polynomial size $(1,+1)$-ReBP-s are more powerful than polynomial size read-once branching programs and that polynomial size $(1,+(k+1))$-ReBP-s are more powerful than polynomial size $(1,+k)$-ReBP-s.

Recall definition of branching programs. An X-input ($X=\{x_1,\dots,x_n\}$) nondeterministic BP (NBP) P for computing a function $g(X)$ ($g:\{0,1\}^n\to\{0,1\}$) is a directed acyclic multigraph with a distinguished source node s and distinguished sink nodes *accept* and *reject*. The out degree of each non-sink node is *at least* 2, all outgoing edges are labeled by $x_i=0$ or $x_i=1$ for variable x_i associated with the node and *at least one* of outgoing edge is labeled by $x_i=0$ and *at least one* of outgoing edge is labeled by $x_i=1$. The label "$x_i=\delta$" indicates that only inputs satisfying $x_i=\delta$ may follow this edge in the computation. NBP P computes a function g in the obvious way. That is, for each $\sigma\in\{0,1\}^n$ we let $g(\sigma)=1$ iff there is a directed path starting in the source and leading to the accepting node such that all labels $x_i=\sigma_i$ along this path are consistent with $\sigma=\sigma_1,\sigma_2,\dots,\sigma_n$.

BP P is *deterministic* (DBP) if the out degree 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. Note that for DBP P each input sequence σ determines unique path consistent with σ.

For BP P we define size(P) (complexity of the branching program P) as the number of internal nodes in P.

Regular $(1,+k)$-BP and one-way communication with

Definition 1 *Call P a regular $(1,+k)$-branching program ($(1,+k)$-ReBP) iff (i) for each node v of P on each path from the source to v the same set $X(v)\subseteq X$ of variables is tested, (ii) on each path from the source to a sink all variables X are tested, and (ii) along every consistent path at most k variables are tested more than once.*

Clearly we have that OBDD-s and oblivious BP-s are regular BP-s. Note that an arbitrary read-once BP P can be transformed into a regular read-once BP P' by inserting dummy tests such that $size(P')\le 2n\,size(P)$. Note that the procedure of inserting dummy tests for ordinary $(1,+k)$-BP P (in order to get regular BP P' from P) can violates the $(1,+k)$ property.

Using the communication complexity approach we prove exponential (in n) lower bound for the size for any nondeterministic $(1,+k)$-ReBP which presents permutation function $PERM_{n^2}$ over n^2 variables. Remind that $PERM_{n^2}$ is polynomially easy for ordinary nondeterministic $(1,+1)$-BP [6]. We also show that nondeterministic $(1,+k)$-ReBP are more powerful than read-once NBP. Note that the problem of proving exponential lower bound for some explicit function presented in nonsyntactic NBP models more powerful than read-once model has been open before our result. Remind that in [9] it was proved that polynomial size read-once NBPs are more powerful than their deterministic counterpart.

2 Results

Sieling defined and investigated the power of OBDD$_{+k}$ and syntactic $(1,+k)$-BP models in [15]. Informally speaking OBDD$_{+k}$ is OBDD which can test up to k variables after first reading of all his variables. Note that different paths of OBDD$_{+k}$ can test different k extra variables. It is proved in [15] that polynomial size OBDD$_{+(k+1)}$-s are more powerful than polynomial size OBDD$_{+k}$-s.

Clearly we have that for $k = O(\log n)$ X-input, $(|X| = n)$ polynomial size OBDD$_{+k}$ can be transformed to X-input polynomial size $(1,+k)$-ReBP.

The Boolean function $f_{n,k}$ is defined in [14]. Informally it is defined as follows. The n variables X are divided into k blocks of length m. For every $j = 1, 2, \ldots, k$ a weighted sum of the bits of block j determines an index i_j of input bits. Then, the value of the function is the parity of bits determined by i_j for $j = 1, 2, \ldots, k$. See [14] for the formal definition.

Savicky and Zak proved [14] that $f_{n,k}$ needs exponential size for presentation by $(1,+(k-1))$-DBP. Using the method of the paper [15] we can construct $(1,+k)$-ReBP (in the form OBDD$_{+k}$) for presentation $f_{n,k}$ of size $O(n^k)$. This proves proper hierarchy of the computational power of $(1,+k)$-ReBP-s in respect of parameter k . In particular this proves that $(1,+k)$-ReBP-s are more powerful than read-once BP-s.

Permutation ($PERM_{n^2}$) function (or *exact-perfect-matching* function) investigated by different authors (see for example [4, 9]). Given $n \times n$ Boolean matrix A. One has to determine whether A is a permutation matrix. That is, whether there is precisely one 1 in every row and every column.

For $q \in [0,1]$ denote $H(q)$ the Shannon entropy function. That is, $H(q) = -q \log q - (1-q)\log(1-q)$.

Theorem 1 *For any nondeterministic* $(1,+k)$*-ReBP P that computes $PERM_{n^2}$, for $q = k/n^2$ it holds that*

$$size(P) \geq 2^{(n-n^2H(q)-k)/2}/2\sqrt{n}.$$

The proof of the theorem presented in the section 3.3.

Corollary 1 *Let $k = o(n^2)$. For any nondeterministic* $(1,+k)$*-ReBP P that computes $PERM_{n^2}$ for arbitrary $\varepsilon \in (0,1)$ for n large enough it holds that*

$$size(P) \geq 2^{(n-k)/2-k\log(n^2/k)}/2\sqrt{n}.$$

Proof. For $k = o(n^2)$ we have that $H(q) \sim (k/n^2)\log(n^2/k)$. Using the lower bound of the theorem 1 we get the statement of the corollary. □

Note that $PERM_{n^2}$ is polynomially easy for $(1,+1)$-NBP [5], for deterministic read-2-times branching programs [20] and randomized OBDD [19].

3 Proofs

Below we present two general lower bounds for $(1,+k)$-ReBP based on combination of decomposing of $(1,+k)$-ReBP with overlapping communication complexity technique (note that overlapping communication complexity appeared to be very productive for the investigation VLSI circuits [8]). Next we apply the general lower bounds for proving lower bounds for Theorem 1.

3.1 Decomposition of $(1, +k)$-ReBP

Consider X-input $(1, +k)$-ReBP P. Call an edge (v, v') of P an x_i-edge if it is labeled by $x_i = 0$ or $x_i = 1$.

Path π of P is called a consistent path (see [2]) if it is consistent with some input sequence $\sigma = \sigma_1, \ldots, \sigma_n$. That is, $\pi = (v_0, \ldots, v_l)$, where v_0 is a source of P, v_l is one of a sinks of P and each x_j-edge (v_i, v_{i+1}) of π is labeled by $x_j = \sigma_j$.

Definition 2 *We call a consistent path of BP P a computation of P.*

Let $S \subset X$. Define an S-*border* set $B(S)$ of nodes of P as follows:

$$B(S) = \{v : X(v) = S \textit{ and for all offspring nodes of } v\ S \subset X(v') \textit{ properly } \}.$$

Denote $P(S)$ a subgraph of P (and call it subprogram of P) which is determined by all computations π with the property: *path π contains a node from $B(S)$*. Note that for nondeterministic P its $P(S)$ subprogram can be deterministic.

Property 1 *Each path π of $P(S)$ contains exactly one node from $B(S)$.*

Proof: Evident. □

We will view on $P(S)$ as a BP consisting of two parts P^1 and P^2 where P^1 consists of the part of $P(S)$ "before" $B(S)$, including S-border $B(S)$ and P^2 — is a remind part of $P(S)$.

Formally P^1 and P^2 can be described as follows. Due to Property 1 nodes of S-border set $B(S)$ determines a partition of each computation π of $P(S)$ into two parts π^1 and π^2. That is, if $\pi = (v_0, \ldots, v_i, v_{i+1}, \ldots v_l)$ where $v_i \in B(S)$ then $\pi^1 = (v_0, \ldots, v_i)$ and $\pi^2 = (v_{i+1}, \ldots, v_l)$. Now we define P^1 to be a first part of $P(S)$ determined by first parts π^1 of computations π of $P(S)$ and define P^2 to be a remind part of $P(S)$.

Let $Z \subset X$, $|Z| \leq k$ and let ρ be a map $\rho : Z \to \{0, 1\}$. Denote $P|_\rho(S)$ a subgraph of $P_Z(S)$ with the property:

- for any computation π of $P(S)$ only variables from Z are tested both by the first part P^1 and the second part P^2 of $P(S)$.
- for any computation π all its z_i-edges, $z_i \in Z$, are marked $z_i = \sigma_i$ in according to the map ρ.

Definition 3 *We call a family $R = \{S : S \subset X\}$ of subsets of X a P-family if it is true that each computation π of P belongs to some subprogram $P(S)$ of P for $S \in R$ and removing arbitrary set S from R violates this property.*

The following lemma is motivated by the exposition in [2].

Lemma 1 (decomposing lemma) *Let P be a nondeterministic $(1,+k)$-ReBP. Let R be a P-family. Then P can be represented in the form*

$$P = \bigcup_{S \in R} \bigcup_{\substack{\rho : Z \to \{0,1\}, \\ |Z| \leq k}} P|_\rho(S).$$

Proof: Each $S \in R$ determines a subprogram $P(S)$ of P and each computation π belongs to some subprogram $P(S)$ of P for $S \in R$. So, $P = \bigcup_{S \in R} P(S)$. Each computation π of $P(S)$ uniquely determines a set $Z \subset X$, $|Z| \leq k$, of all variables tested along π both in the first part P^1 and the second part P^2 of $P(S)$ and a map $\rho : Z \to \{0,1\}$. So, $P(S) = \bigcup_{\substack{\rho : Z \to \{0,1\}, \\ |Z| \leq k}} P|_\rho(S)$. □

We call the presentation of $(1,+k)$-ReBP P from Lemma 1 an *R-decomposition of P* or just *decomposition of P*.

Let $\Sigma \subseteq \{0,1\}^{|X|}$ be some selected set of inputs for P. Call $|\Sigma|$ (cardinality of Σ) a Σ-weight of P and denote it $\omega_\Sigma(P)$. Next consider an R-decomposition of P. For a subprogram $P|_\rho(S)$ of R-decomposition of P let $\Sigma' \subseteq \Sigma$ be a set of all inputs which determine computations of $P|_\rho(S)$. Call $|\Sigma'|$ a Σ-weight of $P|_\rho(S)$ and denote it $\omega_\Sigma(P|_\rho(S))$.

Denote MP_R a subprogram from R-decomposition of P of maximum Σ-weight and denote $\omega_\Sigma(MP_R)$ its Σ-weight. That is,

$$\omega_\Sigma(MP_R) = \max_{S \in R} \max_{\substack{\rho : Z \to \{0,1\}, \\ |Z| \leq k}} \{\omega_\Sigma(P|_\rho(S))\}.$$

Lemma 2 *Let P be an X-input $(1,+k)$-ReBP and let $\Sigma \subseteq \{0,1\}^{|X|}$ be some selected set of inputs for P. Let $q = k/|X| \leq 1/2$. Then for arbitrary P-family R the following is true*

$$\omega_\Sigma(MP_R) \geq |\Sigma| / \left(2^{|X|H(q)+k} size(P)\right).$$

Proof: From Lemma 1 it follows that the sum of Σ-weights of all subprograms of R-decomposition of P is at least $\omega_\Sigma(P)$ (exactly $\omega_\Sigma(P)$ when P is deterministic)

$$|\Sigma| = \omega_\Sigma(P) \leq \sum_{S \in R} \sum_{\substack{\rho : Z \to \{0,1\}, \\ |Z| \leq k}} \omega_\Sigma(P|_\rho(S)).$$

Observe that for $S, S' \in R$, $S \neq S'$, from the regularity property of P it follows that $B(S) \cap B(S') = \emptyset$. From this we get that

$$|R| \leq \sum_{S \in R} |B(S)| \leq size(P).$$

Now from the above two inequalities it follows that

$$|\Sigma| \leq \sum_{S \in R} \sum_{\substack{\rho : Z \to \{0,1\}, \\ |Z| \leq k}} \omega_\Sigma(P|_\rho(S)) \leq size(P) \left(\sum_{i=1}^{k} \binom{|X|}{i} \right) 2^k \omega_\Sigma(MP_R).$$

Using Chernoff upper bound $\sum_{i=1}^{k} \binom{|X|}{i} \leq 2^{|X|H(q)}$ for the case $q = k/|X| \leq 1/2$ we get the statement of the lemma. □

3.2 Lower bounds for $(1,+k)$-ReBP

Let $MP = P|_\rho(S)$. Let $B(S)$ be an S-border set of $P|_\rho(S)$. We reduce a problem of proving lower bound for $size(P)$ for proving a lower bound for $|B(S)|$.

For estimating a lower bound for $|B(S)|$ we apply two combinatorial methods based on:
(i) "communication" technique used for ordered and oblivious models of BP-s [1, 3, 10] and
(ii) "weights" technique based on that of [9]. This technique was generalized in [18, 2].

Denote ψ a Boolean function computable by subprogram $P|_\rho(S)$ of P. Note that for DBP P that computes function f for all inputs γ of $P|_\rho(S)$ it holds that $f|_\rho(\gamma) = \psi(\gamma)$ but for NBP P this can be not true (for an input γ for which $f|_\rho(\gamma) = 1$ the subprogram $P|_\rho(S)$ can contain only rejecting computations of P).

Communication technique. Denote $S_1 = S\backslash Z$, $S_2 = X\backslash(S_1 \cup Z)$. Subprogram $P|_\rho(S)$ of P is a *weak ordered.*

Consider an $(S_1 : S_2)$-communication computation of ψ based on that of Yao [21]. Two players A (Alice) and B (Bob) wish to compute ψ. A gets all bits in S_1 and B — all in S_2. A starts a computation, B on obtaining a message from A finishes the computation and outputs the result. Denote $NC^{S_1:S_2}(\psi)$ $(DC^{S_1:S_2}(\psi))$ a nondeterministic (deterministic) one-way communication complexity of computing ψ described above. Denote $CM^{S_1:S_2}(\psi)$ a communication matrix. That is, $CM^{S_1:S_2}(\psi)$ is an $|S_1| \times |S_2|$ Boolean matrix, (S_1, S_2) entry of $CM^{S_1:S_2}(\psi)$ is $\psi(S_1, S_2)$. Denote $nrow(CM^{S_1:S_2}(\psi))$ to be a number of different rows of $CM^{S_1:S_2}(\psi)$.

Lemma 3 *Let P be an X-input $(1,+k)$-ReBP. Let $P|_\rho(S)$ be an arbitrary subprogram of decomposition of P and ψ be a function computable by $P|_\rho(S)$. If P is nondeterministic then*

$$|B(S)| \geq 2^{NC^{S_1:S_2}(\psi)}.$$

If P is deterministic then

$$|B(S)| \geq nrow(CM^{S_1:S_2}(\psi)).$$

Proof: Describe the following communication protocol Φ, which computes function ψ. Let $\gamma \in \Sigma$ be a valuation of X. Denote $\gamma = (\sigma^1; \sigma; \sigma^2)$, where $\sigma^1 \mapsto S_1$, $\sigma \mapsto Z$, $\sigma^2 \mapsto S_2$ assignments of γ to S_1, S_2, and Z respectively.

Players A and B receive respectively σ^1 and σ^2. Setting $\sigma \mapsto Z$ is known both to A and B. This model of computation is known as a communication computation with overlap information [7, 11]. Let $v_1, \ldots, v_d$ be all internal nodes of $P|_\rho(S)$ that are reachable during paths of computation on the part σ^1 of γ.

During the computation on the input σ^1 player A nondeterministically selects and sends node v_i to player B. Player B on obtaining message v_i from A starts its computation (simulation of $P|_\rho(S)$) from the node v_i on the part σ^2 of the input γ.

The statements of the lemma result from the definition of the protocol Φ and known fact that $DC^{S_1:S_2}(\psi) = \lceil \log nrow(CM^{S_1:S_2}(\psi)) \rceil$ [21]. □

Weights technique. Let Σ be some subset of inputs, $\Sigma \subseteq \{0,1\}^{|X|}$ of P. Consider R-decomposition of P. Let $P|_\rho(S)$ be a subprogram from R-decomposition of P and let $B(S)$

be an S-border set of $P|_\rho(S)$. The set Σ of inputs of P determines a set Σ' of inputs of $P|_\rho(S)$. For $v \in B(S)$ we define $\Sigma(v)$ as follows:

$$\Sigma(v) = \{\gamma \in \Sigma' : \textit{ exists } \pi_\gamma \textit{ that goes through } v \ \}.$$

Denote

$$\omega^*_\Sigma(P|_\rho(S)) = \max\{|\Sigma(v)| : v \in B(S)\}.$$

Lemma 4 *Let P be a nondeterministic X-input $(1,+k)$-ReBP. Let $P|_\rho(S)$ be an arbitrary subprogram of decomposition of P. Then for arbitrary subset Σ of $\{0,1\}^{|X|}$ it holds that*

$$|B(S)| \geq \omega_\Sigma(P|_\rho(S))/\omega^*_\Sigma(P|_\rho(S)).$$

Proof: From the definition of $B(S)$ it follows that $\bigcup_{v\in B(S)} \Sigma(v) = \Sigma'$. This means that $\sum_{v\in B(S)} |\Sigma(v)| \geq |\Sigma'|$ or

$$|B(S)|\omega^*_\Sigma(P|_\rho(S)) \geq \omega_\Sigma(P|_\rho(S)).$$

□

Clearly we have that in our case the weight technique is a variant of communication method. That is, in the case of one-way communication game ω^* is the maximum number of inputs which initialize the same message of player A to player B. So, the number ω of different inputs divided by ω^* gives the lower bound to the number of messages used by communication protocol.

3.3 Proof of Theorem 1

Let P be a $(1,+k)$-ReBP P that computes $PERM_{n^2}$.

Part 1: Consider Σ to be a set of ones of $PERM_{n^2}$ that is, $\Sigma = PERM^{-1}_{n^2}(1)$, $|\Sigma| = n!$

Part 2: Let $m = n/2$. For $\gamma \in \Sigma$ denote π_γ an accepting computation of P consistent with γ. for accepting computation π_γ denote π^1_γ a first part of π_γ such that exactly m edges of π^1_γ marked by 1 (say that an edge (v,v') is marked by 1 if it is labeled by $x_i = 1$ for some $x_i \in X$). Denote $S(\pi^1_\gamma)$ a set of all variables tested along π^1_γ. Now define the P-family R as follows.

$$R = \{S \subset X : S = S(\pi^1_\gamma), \gamma \in \Sigma, \pi^1_\gamma \text{ — is the accepting path } \}.$$

Consider R-decomposition of P and subprogram MP from R-decomposition of P. Remind that $MP = P|_\rho(S)$ is a subprogram from R-decomposition of P of maximum Σ-weight. For $q = k/n^2$ from Lemma 2 it follows that

$$\omega_\Sigma(MP) \geq n!/\left(2^{n^2H(q)+k} size(P)\right). \tag{1}$$

Part 3: Use the "weight" technique.

Lemma 5 *For MP it holds that*

$$\omega^*_\Sigma(MP) \leq (m!)^2.$$

Proof: Denote Σ' set of all inputs from Σ which determine computations of $MP = P|_\rho(S)$. Input $\gamma \in \Sigma'$ determines permutation $\Pi_\gamma = \left(\begin{array}{ccc} 1 & \dots & n \\ i_1 & \dots & i_n \end{array}\right)$. Due to the map $\rho : Z \to \{0,1\}$ the part Π_γ^Z of Π_γ determined by the set Z is the same for all $\gamma \in \Sigma'$. From this using known arguments (see for example [9]) we get that

$$\omega^*_\Sigma(MP) \leq ((m-l)!)^2 \leq (m!)^2.$$

□

Now from the lemma above, Lemma 4, and from (1) we get the following lower bound

$$|B(S)| \geq 2^{n-n^2H(q)-k}/(2\sqrt{n}\, size(P)),$$

or

$$size(P)^2 \geq 2^{n-n^2H(q)-k-\log\sqrt{n}}/2.$$

The last in equation proves the lower bound of the theorem 1.

Concluding remarks

In the paper we used only technique described in part "weight technique" and did not use the technique described in part "communication technique" for proving lower bounds. Note, that (as an example of using explicitly the "communication technique") one can directly prove exponential lower bound (known from [14]) for presentation $f_{n,k}$ by deterministic $(1, +(k-1))$-ReBP.

Detlef Sieling in his comments for the draft version of the paper mentioned that he think that lower bound method from [15] can be adapted for regular (1,+k)-BPs. In this context also the results of the paper [16] might be interesting.

It is an interesting open problem to prove exponential lower bound for complexity of presentation multiplication function by $(1, +k)$-ReBP using the communication complexity technique.

Acknowledgments. I would like to thank Ingo Wegener, Detlef Sieling, Sasha Razborov and members of complexity seminars in Steklov Mathematical Institute and Dortmund University for the helpful discussion on the subject of the paper. I am grateful to Detlef Sieling for his comments on the draft of the paper.

References

[1] F. Ablayev and M. Karpinski, On the Power of Randomized Ordered Branching Programs, *Electronic Colloquium on Computational Complexity*, TR98-004, (1998), available at `http://www.eccc.uni-trier.de/eccc/`

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

[3] R. Bryant, On the complexity of VLSI implementations and graph representations of Boolean functions with applications to integer multiplication, *IEEE Trans. Comput.*, 40 (2), (1991), 205-213.

[4] S. Jukna, The effect of null-chains on the complexity of contact schemes, *in Proceedings of the 7th FCT, Lecture Notes in Computer Science, Springer-Verlag*, 380, (1989), 246-256.

[5] S. Jukna, A note on read-k-times branching programs, *RAIRO Theoretical Informatics and Applications*, **29**:1, (1995), 75-83.

[6] S. Jukna, Complexity of Boolean Functions, see *Electronic Colloquium on Computational Complexity*, section Lecture Notes, available at `http://www.eccc.uni-trier.de/eccc/`

[7] Hromkovic J, Communication Complexity and Parallel Computing, *EATCS Series, Springer-Verlag,* (1997), 336 p.

[8] Hromkovic J, Communication Complexity and Lower Bounds on Multilective Computations, *Theoretical Informatics and Applications*, 33, (1999), 193-212; see also the extended abstract of the paper *in trans. of 23rd MFCS'98, Lect.Notes in Comp. Sci, Springer-Verlag*, 1450, (1998), 789-797.

[9] M. Krause, C. Meinel, and S. Waack, Separating the eraser Turing machine classes L_e, NL_e, and P_e, *Theoretical Computer Science*, 86, (1991), 267-275.

[10] M. Krause, *Lower Bounds for Depth-Restricted Branching Programs*, Information and Computation, 91, (1991), pp. 1-14.

[11] E. Kushilevitz and N. Nisan, Communication complexity, *Cambridge University Press*, 1997.

[12] S. Ponzio, Restricted branching programs and hardware verification, *PHD theses, MIT*, (1995), available at `http://www.eccc.uni-trier.de/eccc/`

[13] 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.

[14] 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/`

[15] D. Sieling, New lower bounds and hierarchy results for restricted branching programs, *J. of Computer and System Sciences.* Vol. 53, No. 1, (1996), pp. 79-87 See also *Electronic Colloquium on Computational Complexity*, TR95-002, (1995), available at `http://www.eccc.uni-trier.de/eccc/`

[16] D. Sieling, A separation of syntactic and nonsyntactic (1,+k)-branching programs, *Electronic Colloquium on Computational Complexity*, TR98-045, (1998), available at `http://www.eccc.uni-trier.de/eccc/`

[17] D. Sieling and I. Wegener, New lower bounds and hierarchy results for restricted branching programs, In *Proc. of Workshop on Graph-Theoretic Concepts in Computer Science WG'94, Lect. Notes in Comp. Sci., Springer-Verlag*, 903, (1994), 359-370.

[18] J. Simon and M. Szegedy, Anew lower bound theorem for read-only-once branching programs and its applications, *DIMAC Series in Discrete Mathematics and Theoretical Computer Science*, 13, (1993), 183-193.

[19] M. Sauerhoff, A lower bound for randomized read-k-times branching programs, *in Proceedings of STACS'98,* LNCS (1998), 1373, pp. 105-115, see also *Electronic Colloquium on Computational Complexity*, TR97-019, (1997), available at `http://www.eccc.uni-trier.de/eccc/`

[20] I. Wegener, The complexity of Boolean functions, Wiley-Teubner, 1987.

[21] 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.