Electronic Colloquium on Computational Complexity, Report No. 37 (2001)

Bounded-Width Probabilistic OBDDs and Read-Once Branching Programs are Incomparable

Rustam Mubarakzjanov *

Abstract

Restricted branching programs are considered by the investigation of relationships between complexity classes of Boolean functions. Read-once ordered branching programs (or *OBDD*s) form the most restricted class of this computation model. Since the problem of proving exponential lower bounds on the complexity for general probabilistic *OBDD*s is open so far, it is interesting to study this problem in a restricted setting. For this reason we deal in this work with probabilistic *OBDD*s whose width is bounded.

We prove in this work that probabilistic *OBDD*s of width bounded by a constant can be more powerful than even non-deterministic read-once branching programs. To do it we present a probabilistic *OBDD* of constant width computing the known function $PERM$. We prove for several known functions that they cannot be computed by probabilistic *OBDD*s of constant width. To show it we present a new method allowing to obtain lower bound $\Omega(n)$ on the width of corresponding *OBDD*s (n is the number of variables).

1 Introduction

In order to study the relationship between different complexity classes restricted models of computation are considered. Branching programs are one of the most investigated computation models (see [17], [5] for a lot of references) during the last years. In particular, read-once ordered branching programs (*OBDD*s) determine complexity classes whose relationships

*Dept. of Computer Science, University of Trier. Research partially supported by the Russia Fund for Basic Research 99-01-00163 and by the Grant "Russia Universities" 04.01.52. Email: **rustam@ti.uni-trier.de**

ISSN 1433-8092

are successfully proven [2], [11], [12]. The same time *OBDD*s are convenient tools to represent Boolean functions because of the possibility to manipulate them efficiently [7]. Probabilistic *OBDD*s are the most general *OBDD*s.

We recall basic definitions. A *deterministic* branching program P is a directed acyclic multi-graph with a source node and two distinguished sink nodes: accepting and rejecting. The out-degree of each non-sink node is exactly 2. Each node is labeled by some variable x_i, and two arcs outgoing from x_i-node are labeled by 0 and 1. The label "a" indicates that only inputs satisfying $x_i = a$ may follow this arc in the computation. A branching program P computes a function h_n in the obvious way: for each $\mathbf{x} \in \{0,1\}^n$, $h_n(\mathbf{x}) = 1$ if and only if there is a directed path starting in the source and leading to the accepting node such that all labels along this path are consistent with $\mathbf{x} = x_1 x_2 \dots x_n$. The branching program becomes *non-deterministic* if we allow guessing nodes, that are nodes with two outgoing arcs being unlabeled. A non-deterministic branching program P outputs 1 on an input $\mathbf{x}$ if and only if there exists (at least one) computation on $\mathbf{x}$ starting in the source node and leading to the accepting node. A *probabilistic* branching program has in addition to its standard (deterministic) nodes specially designated nodes called random nodes. Each such node corresponds to a random input y_i having values from $\{0,1\}$ each with probability $1/2$. We say that a probabilistic branching program *computes* a function h if it outputs $h(\mathbf{x})$ with probability greater than $1/2$ for any input $\mathbf{x}$. If this probability is at least $1/2 + \epsilon$ for some $\epsilon > 0$ one says that the computation has bounded error ϵ. A probabilistic branching program P on n variables determines a function c_P: $\{0,1\}^n \to [0,1]$; $c_P(\mathbf{x})$ is the probability that P reaches the accepting sink on the input $\mathbf{x}$. We call this function the *characteristic function* of the branching program P.

We define the *complexity* of a branching program P as the number of its nodes. We denote the class of Boolean functions computable by polynomial size nondeterministic branching programs as *NP-BP*. We say that a function belongs to the class *PP-BP* if and only if there is a polynomial size probabilistic branching program computing this function. For a probabilistic computation with bounded error, we use another notation for the complexity class. Let BPP_ϵ-BP be the class of functions computable with error $\epsilon > 0$ by polynomial size probabilistic branching programs. Furthermore, let $BPP\text{-}BP := \bigcup_{0<\epsilon\leq 1/2} BPP_\epsilon\text{-}BP$. For a restricted class of branching programs Q, we define analogous complexity classes using "-Q" as a suffix to their notations.

A *read-once* branching program (*BP1*) is a branching program in which every variable is tested at most once on each path. A *BP1* is called *ordered* (or *OBDD*) if the variables have to be tested according to some fixed ordering π. An *OBDD* is called *oblivious* if it can be leveled, i.e. arcs lead only to nodes of the neighboring level, and each level contains only x-nodes for some fixed variable x. The *width* of an *OBDD* is the maximum number of nodes belonging to a level. *OBDD*s having the width bounded by a constant (*bwOBDD* for short) are studied in this work. The restriction on the width was studied earlier for general deterministic branching programs [3]. Recently, Newman [16] showed that functions computable by deterministic *bwOBDD*s can be computed probabilistically with bounded error with constant number of queries.

Although there are results concerning incomparability of probabilistic *OBDD*s with bounded error on the one hand and non-deterministic *OBDD*s [1] or even non-deterministic *BP1*s [19] on the other hand, the power of probabilistic *OBDD*s without bounded error was not studied yet. In this paper we present a new method helping to find probabilistic *OBDD*s computing certain functions and a new technique to find lower bounds on the width of probabilistic *OBDD*s. We show in this work that probabilistic *bwOBDD* can be more powerful than non-deterministic *BP1*. We present basic lemmas helping to obtain different characteristic functions of some probabilistic *bwOBDD*s. Using these lemmas, we prove that the function *PERM* known to be hard for non-deterministic read-once branching programs can be computed by probabilistic *bwOBDD*.

There is no known exponential lower bound of the complexity of probabilistic *OBDD*s without bounded error. We prove for several known functions that they can not be computed by probabilistic *OBDD*s of a constant width. To show it we present a new method allowing to give lower bound $\Omega(n)$ on the width of corresponding *OBDD*s (n is the number of variables).

2 Characteristic functions of probabilistic bwOBDDs

What kind of functions can be the characteristic functions of *OBDD*s or of *bwOBDD*s? We need the following modifications of simple lemmas from [14] (see also [15]).

Lemma 1 *For any constant α, $0 \leq \alpha \leq 1$, if the binary representation of α has t positions then there exists a bwOBDD $B(\alpha)$ of width 2 with t levels consisting only of random nodes such that $c_{B(\alpha)} = \alpha$.*

Lemma 2 *Let c_{B_1} and c_{B_2} be the characteristic functions of bwOBDDs B_1*

of width w_1 *and* B_2 *of width* w_2*, respectively, reading deterministic variables of the same set* X *in the same order. Then the following functions are the characteristic functions of some bwOBDDs with the same variable order:* $1 - c_{B_1}(\mathbf{x})$*,* $1/2(c_{B_1}(\mathbf{x}) + c_{B_2}(\mathbf{x}))$*,* $c_{B_1}(\mathbf{x})c_{B_2}(\mathbf{x})$*. These OBDDs that we denote as* $1 - B_1$*,* $1/2(B_1 + B_2)$*, and* B_1B_2 *have the width* w_1*,* $w_1 + w_2$*, and* w_1w_2 *respectively.*

If bwOBDDs B_1 *and* B_2 *have disjoint sets of variables then there are bwOBDDs* $1/2(B_1+B_2)$ *and* B_1B_2 *with characteristic functions* $1/2(c_{B_1}(\mathbf{x})+c_{B_2}(\mathbf{x}))$ *and* $c_{B_1}(\mathbf{x})c_{B_2}(\mathbf{x})$ *of width* $max(w_1, w_2) + 1 - sg(|w_1 - w_2|)$ *and* $max(w_1, w_2)$ *respectively.*

Using these lemmas it is easy to construct a *bwOBDD* computing some function f_n if a *bwOBDD* B with the following property is known. There is some number p_n such that $c_B(\mathbf{x}) > p_n$ if and only if $f_n(\mathbf{x}) = 1$. The following theorem presents a method to produce a desired *bwOBDD* if for a known *bwOBDD* B and some number p_n, $c_B(\mathbf{x}) = p_n$ if and only if $f_n(\mathbf{x}) = 1$.

Theorem 1 *Let* f_n *be a function on* n *variables. Let* p_n *be a real and* B *be a probabilistic OBDD of width* c *with the following property. For every word* $\mathbf{x}, |\mathbf{x}| = n$*,* $f_n(\mathbf{x}) = 1$ *if and only if* $c_B(\mathbf{x}) = p_n$*. Then* f_n *is computable by a probabilistic OBDD of width* $c^2 + 2c + 2$*.*

Proof. Let B have n' levels. Then there is an $\epsilon_n \geq (1/2)^{n'}$ such that if $f(\mathbf{x}) = 0$ then $|c_B(\mathbf{x}) - p_n| \geq \epsilon_n$. Let s_n be a real with binary representation equal to the prefix of $n' + 1$ bits of the binary representation of p_n: i.e. $p_n = 0.p_n^{(1)} \dots p_n^{(n'+1)} \dots$ and $s_n = 0.p_n^{(1)} \dots p_n^{(n'+1)}$.

Consider $s_n > 1/2$. Because of Lemmas 1 and 2, there exists a *bwOBDD* $B_1 = B(\frac{1}{2}(1 - B + B(p')))$ of width $c(c+2)$ with the characteristic function $c_{B_1} = c_B(\frac{1}{2}(1 - c_B + p'))$, $p' = 2s_n - 1$. The function c_{B_1} has the maximum equal to $p'' = s_n^2/2$ if $c_B = s_n$.

If $s_n \leq 1/2$ then there exists a *bwOBDD* $B_1 = (1 - B)(\frac{1}{2}(B + B(p')))$ of width $c(c + 2)$ with the characteristic function $c_{B_1} = (1 - c_B)(\frac{1}{2}(c_B + p'))$, $p' = 1 - 2s_n$. The function c_{B_1} has the maximum equal to $p'' = (1 - s_n)^2/2$ if $c_B = s_n$.

For both cases if $c_B(\mathbf{x}) \neq p_n$, i.e. $f_n(\mathbf{x}) = 0$, then $c_{B_1} \leq p'' - \epsilon_n^2/2$ otherwise $f_n(\mathbf{x}) = 1$ and $c_{B_1} \geq p'' - \frac{1}{2}^{(2(n'+1)+1)}$. Let p''' be a real with binary representation equal to the prefix of $2n' + 3$ bits of the binary representation of $1 - p'' + \epsilon_n^2/4$. The *bwOBDD* $1/2(B_1 + B(p'''))$ of width $c(c+2) + 2$ with the characteristic function $\frac{1}{2}(c_{B_1} + p''')$ computes the function f_n. ∎

We studied in this section $OBDD$s with a fixed variable ordering. Although these computation seem to be somewhat poor they are sufficiently powerful for our purpose.

3 Probabilistic OBDD of constant width can do more than polynomial nondeterministic BP1s

The author presented in [14] a function belonging to the class $Q = PP\text{-}bwOBDD \setminus (BPP\text{-}OBDD \cup NP\text{-}OBDD)$. Now we investigate the even harder function $PERM_n$ corresponding to the set of *permutation matrixes*. It is known ([9], [13]) that this function is hard for nondeterministic read-once branching programs. Recall that $PERM_n : \{0,1\}^{m^2} \to \{0,1\}, n = m^2$, and $PERM_n(\mathbf{x}) = 1$ if and only if every row and every column of the Boolean $m \times m$-matrix $x = (x_{1,1}, x_{1,2}, \ldots, x_{m,m})$ contains exactly one 1. The function $PERM_n$ can be computed by a polynomial size probabilistic $OBDD$ with bounded error [18]. Due to [14] probabilistic $bwOBDD$s with bounded error are not more powerful than deterministic $bwOBDD$s. Therefore, $PERM_n$ does not belong to $BPP\text{-}bwOBDD$. We show that this function can be computed by a probabilistic $bwOBDD$ if there is no bound on the error.

Theorem 2 *The function $PERM_n$ can be computed by a probabilistic $OBDD$ $B(PERM_n)$ of constant width.*

Proof. We describe the $OBDD$ $B(PERM_n)$ computing $PERM_n$. This $OBDD$ reads variables in the order $x_{1,1}, x_{1,2}, \ldots, x_{m,m}$. $B(PERM_n)$ has the following parts. For any i, $1 \le i \le m$, a deterministic $OBDD$ $P_1^{(i)}$ and a probabilistic $OBDD$ $P_2^{(i)}$ read the i-th row of the matrix and check whether this row contains exactly one 1. If this is the case then $P_2^{(i)}$ reaches the accepting sink with probability $(\frac{1}{2})^j$ for $x_{i,j} = 1$. There exist such $OBDD$s having on each level 3 nodes one of which will be called *rejecting* node. All paths from this rejecting node lead to the rejecting sink.

We firstly describe a probabilistic $OBDD$ P for which reals p_m and ϵ_m exist such that $c_P(\mathbf{x}) = p_m$ if $PERM_n(\mathbf{x}) = 1$, and $|p_m - c_P(\mathbf{x})| \ge \epsilon_m$ otherwise. The source of P is a random node selecting $P_2^{(1)}$ or $P_1^{(1)}$.

The accepting sink of $P_1^{(i)}$, $1 \le i < m-1$, is identified with a random node leading to $P_1^{(i+1)}$ or to $P_2^{(i+1)}$. This part can be written as $P_1^{(i)}(\frac{1}{2}(P_1^{(i+1)} + P_2^{(i+1)}))$. The accepting sink of $P_1^{(m-1)}$ is identified with a random node leading to $P_2^{(m)}$ and to a rejecting node (the subprogram $P_1^{(m-1)}\frac{1}{2}P_2^{(m)}$).

The accepting sink of $P_2^{(i)}$, $1 \leq i < m$, is identified with a random node leading to a rejecting node and to $P_3^{(i+1)}$ (the subprogram $P_2^{(i)}\frac{1}{2}P_3^{(i+1)}$).

The later program $P_3^{(i)}$, $2 \leq i \leq m$, is the copy of $P_1^{(i)}$ which accepting sink, for $i < m$, is identified with a random node leading to a rejecting node and to $P_3^{(i+1)}$ (the subprogram $P_1^{(i)}\frac{1}{2}P_3^{(i+1)}$). Note that $P_3^{(i+1)}$ is reachable from $P_2^{(i)}$ too.

P reaches the accepting sink only if each i-th row of X contains exactly one non-zero element x_{i,j_i} and $c_P(\mathbf{x}) = (\frac{1}{2})^m \sum_{i=1}^{m} (\frac{1}{2})^{j_i}$ in this case. It is easy to see that the function $c_P(\mathbf{x})$ is equal to

$$\left(\frac{1}{2}\right)^m \sum_{i=1}^{m} \left(\frac{1}{2}\right)^i = \left(\frac{1}{2}\right)^{m-1} - \left(\frac{1}{2}\right)^{2m} = p_m$$

if and only if $\{j_i | 1 \leq j_i \leq m, 1 \leq i \leq m\} = \{i | 1 \leq i \leq m\}$ i.e. x corresponds to a permutation matrix. Otherwise $|p_m - c_P(\mathbf{x})| \geq (\frac{1}{2})^{2m}$. Therefore there exists the desired probabilistic *OBDD* $B(PERM_n)$ of constant width (Theorem 1). ∎

Corollary 1 *The function $PERM_n$ is computable by a probabilistic OBDD of width 59.*

Indeed using the construction in the proof of Theorem 1 we transform the *bwOBDD* P having on each level 7 nodes with a rejecting node. Each level of the probabilistic *OBDD* $B(PERM_n)$ has the unique rejecting node. Therefore $B(PERM_n)$ has the width $7(6+2)+2+1=59$.

4 Lower bounds on the width of probabilistic OBDDs

It is known that lower bounds on the complexity of *OBDD*s can be obtained from lower bounds for one-round communication complexity. One considers in such an investigation the following matrix. Let X_1 be a subset of variables set X. Then for a function f on X denote as $CM_f^{X_1}$ the *communication matrix*. Each row of this matrix corresponds to an assignment a to X_1 and each column to an assignment b to $X \setminus X_1$.The element $CM_f^{X_1}(a;b)$ of $CM_f^{X_1}$ on the intersection of this row and this column is equal to $f(a;b)$:

$(a; b)$ denotes in our paper the complete assignment to X where, for example, variables in X_1 obtain assignments corresponding to a.

We shall show that there are functions that can not be computed by probabilistic *bwOBDD*s. These functions are functions known to be hard for randomized *OBDD*s too ([11], [20]). Some of these functions are so-called k-stable functions. But k-stable functions are hard for read-once branching programs too (see e.g., [10]). Because there are functions in the class $P\text{-}BP1 \setminus PP\text{-}bwOBDD$ weaker property than k-stability can be sufficient for our purpose. The same statement holds if one considers $BPP\text{-}OBDD$: not only k-stable functions do not belong to $BPP\text{-}OBDD$. Therefore, for example, Sauerhoff [20] examined by an investigation of the complexity of a function in the context of probabilistic *OBDD*-computations whether this function can be reduced from the known function $INDEX_m$ being hard for one-round communication games. We show directly that functions satisfying a generalized version of k-stability are hard for probabilistic *bwOBDD*s without bounded error.

A Boolean $m \times n$ matrix is *full* if its rows contain every vector from $\{0,1\}^n$. A Boolean matrix is k*-full* if it has a $2^k \times k$ submatrix being full. For example the communication matrix of the function $INDEX_m$ is m-full. If the communication matrix $CM_f^{X_1}$ is k-full then assignments corresponding to the rows and the columns of a $2^k \times k$ full submatrix of $CM_f^{X_1}$ are called *critical.*

Definition 1 *Let f be a function on a set X of n variables. We call this function (k_1, k_2)-***indefinite** *if for any $X_1 \subseteq X$, $|X_1| = k_1$, the communication matrix $CM_f^{X_1}$ is k_2-full.*

To understand the relationship between our definition and the definition of k-stability we just note that if a function is k-stable then it is (k, k)-indefinite.

Theorem 3 *A probabilistic oblivious OBDD computing a (k_1, k_2)-indefinite function f has width at least k_2.*

Proof. Let f be computable by some probabilistic oblivious $OBDD$ P. Consider the level L of P when exactly k_1 variables from X are read. Let this set of read variables be X_1. Let L have $k_2 - 1$ nodes.

Let $\{b_i | 1 \le i \le k_2\}$ and $A = \{a_i | 1 \le i \le 2^{k_2}\}$ be sets of critical assignments to $X \setminus X_1$ and to X_1 respectively. For each $a_i \in A$, we consider the probabilistic distribution $\mu(a_i) = (\mu^{(1)}(a_i) \ldots \mu^{(k_2-1)}(a_i))$ to reach the nodes of L: $\mu^{(j)}(a_i)$ is the probability to reach the j-th node of L from the source of P if the variables from X_1 have values a_i. We also consider for each b_i, $1 \leq i \leq k_2$, a column vector $\nu(b_i) = (\nu^{(1)}(b_i) \ldots \nu^{(k_2-1)}(b_i))$: $\nu^{(j)}(b_i)$ is the probability to reach the accepting sink of P from the j-th node of L if the values of variables from $X \setminus X_1$ correspond to b_i. Then

$$CM_f^{X_1}(a_i; b_j) = 1 \leftrightarrow c_P(a_i; b_j) = \mu(a_i)\nu(b_j) > 1/2.$$

There are coefficients α_i not all equal to 0 such that $\sum_{i=1}^{k_2}(\alpha_i \nu(b_i)) = 0$. Without loss of generality, assume that $\alpha_i \neq 0$ for all $i, 1 \leq i \leq k_2$, and assume

$$\nu(b_1) = \sum_{i=2}^{k_2}(\alpha_i \nu(b_i)).$$

We consider two cases: $\sum_{i=2}^{k_2} \alpha_i \leq 1$ and $\sum_{i=2}^{k_2} \alpha_i > 1$. We take $a \in A$ such that for the first case, $CM_f^{X_1}(a; b_i) = 1$ if $\alpha_i < 0$, $CM_f^{X_1}(a; b_i) = 0$ if $\alpha_i > 0$, where $2 \leq i \leq k_2$, and $CM_f^{X_1}(a; b_1) = 1$. Then

$$1/2 < c_P(a; b_1) = \mu(a)\nu(b_1) = \sum_{i=2}^{k_2}(\alpha_i \mu(a)\nu(b_i)) < 1/2 \sum_{i=2}^{k_2}(\alpha_i) \leq 1/2.$$

If $\sum_{i=2}^{k_2}(\alpha_i) > 1$ then we take $a \in A$ such that $CM_f^{X_1}(a; b_i) = 1$ if $\alpha_i > 0$, $CM_f^{X_1}(a; b_i) = 0$ if $\alpha_i < 0$, $2 \leq i \leq k_2$, and $CM_f^{X_1}(a; b_1) = 0$. Then

$$1/2 > c_P(a; b_1) = \mu(a)\nu(b_1) = \sum_{i=2}^{k_2}(\alpha_i \mu(a)\nu(b_i)) > 1/2 \sum_{i=2}^{k_2}(\alpha_i) > 1/2.$$

It is possible to find this a because $\{b_i | i = 1, k_2\}$ and $A = \{a_i | i = 1, 2^{k_2}\}$ are the sets of critical assignments to $X \setminus X_1$ and to X_1 respectively. The obtained contradictions prove the theorem. ∎

Corollary 2 *The k-stable functions do not belong to PP-bwOBDD.*

The following functions are well investigated: ISA_n, ACH_n, and HWB_n (see for the definitions [6], [4], and [8] respectively). It is known that these functions belong to *P-BP*1 and to *NP-OBDD* $\subset$ *PP-OBDD* ([2]).

Corollary 3 *Let $n = 2^r + r$ and $n = 2m + \log m$. The functions on n variables ISA_n, ACH_n, and HWB_n require a probabilistic OBDDs without bounded error of width $k = \frac{2^r}{r} - 1$, $m/4$, and $0.1n$ respectively.*

We omit here simple proofs following from the original proofs of [6], [4], and [8] respectively that ISA_n is the (k,k)-indefinite function, ACH_n is the $(m, m/4)$-indefinite function, and $HWB_n$is the $(0.6n, 0.1n)$-indefinite function.

Acknowledgment The author would like to thank Farid Ablayev for helpful discussions on the subject of this paper and Martin Mundhenk for his help by preparing this text.

References

[1] F. Ablayev, *Randomization and nondeterminism are incomparable for ordered read-once branching programs*, Proc. ICALP'97, LNCS 1256, Springer, 1997, pp. 195-202.

[2] F.Ablayev, M.Karpinski, and R.Mubarakzjanov, *On BPP versus NP* $\cup coNP$ *for Ordered Read-Once Branching Programs*, Proc. Randomized Algorithms, Brno, 1998, to appear in Theoretical Computer Science.

[3] D.A.Barington, *Bounded-width polynomial-size branching programs recognize exactly those languages in* NC^1, Proc. of the 18st ASM STOC, 1986, 1-5.

[4] B. Bollig, M. Sauerhoff, D. Sieling, and I. Wegener, *On the Power of Different Types of Restricted Branching Programs*, ECCC TR94-026, 1994, `http://www.eccc.uni-trier.de/ECCC`

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

[7] R.E. Bryant, *Graph-based algorithms for Boolean function manipulation*, IEEE Transactions on Computers 35, 1988, 677-691.

[8] R.E. Bryant, *On the complexity of VLSI implementations and graph representations of Boolean functions with application to integer multiplication*, IEEE Trans. Computers, C-40(2): 205-213, Feb.1991.

[9] S.Jukna,*On the effect of null-chains on the complexity of contact schemes*, Proc. of FCT, LNCS 380, 246-256, 1989.

[10] S.Jukna, A.Razborov, P.Savicky, and I.Wegener,*On P versus $NP \cap co-NP$ for decision trees and read-once branching programs*, ECCC TR97-023, 1997, available at http://www.eccc.uni-trier.de/ECCC/

[11] M.Karpinski and R.Mubarakzjanov, *Some separation problems on randomized OBDDs*, CSIT'99, Moscow, Russia, 1999 http://msu.jurinfor.ru/CSIT99

[12] M.Karpinski and R.Mubarakzjanov, *A note on Las Vegas OBDDs*, ECCC TR99-009, 1999 http://www.ecc.uni-trier.de/eccc/

[13] M.Krause, Ch.Meinel, and S.Waak, *Separating the eraser Turing machine classes $L_e, NL_e, co-NL_e and P_e$*, Proc. of MFCS, Springer-Verlag, 1988, pp. 405-413 (LNCS No. 324)

[14] R.Mubarakzjanov, *On probabilistic OBDDs with constant width*, Proceedings of the Workshop on Computer Science and Information Technologies CSIT'2000, Ufa, Russia, September 18-23, 2000, V.2, pp.62-68.

[15] R.Mubarakzjanov, *Probabilistic OBDDs: on Bound of Width versus Bound of Error*, ECCC TR00-085, 2000.

[16] I.Newman, *Testing of function that have small width branching programs*, Proceedings of FOCS (2000).

[17] A.A. Razborov, *Lower bounds for deterministic and nondeterministic branching programs*, Proc. Fundamentals in Computing Theory, Springer, Lecture Notes in Computer Science, 529, 47-60.

[18] M.Sauerhoff, *A lower bound for randomized read-k-times branching programs*, ECCC TR97-019, 1997 http://www.ecc.uni-trier.de/eccc/

[19] M.Sauerhoff, *Randomness and Nondeterminism are incomparable for Read-Once Branching Programs*, ECCC TR98-018, 1998 http://www.ecc.uni-trier.de/eccc/

[20] M.Sauerhoff, *Complexity theoretical results for randomized branching programs*, PhD thesis, Univ. of Dortmund, Shaker 1999 http://ls2-www.informatik.uni-dortmund.de/ sauerhoff/

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'