A Note on Las Vegas OBDDs

Marek Karpinski* Rustam Mubarakzjanov †

Abstract

We prove that the *error-free* (*Las Vegas*) *randomized* OBDDs are *computationally equivalent* to the deterministic OBDDs. In contrast, it is known the same is not true for Las Vegas read-once branching programs.

1 Preliminaries

In the sequel we are going to use the definitions and notations of [KM98]. We recall the complexity classes determined by branching programs: $P{-}BP$, $NP{-}BP$, $coNP{-}BP$, $BPP{-}BP$. We define analogous classes for $OBDD$s using "$-OBDD''$, and for read-k-times branching programs using $''-BPk''$, respectively, as suffixes.

We need also an additional definition. For a probabilistic branching program B, the following subgraph B' of B is called a *random source branching program* (RSP for short) of B: B' is rooted by a random node v of B and there is no random node w that B contains with an edge (w,v). If $B \neq B'$ we call B' proper RSP of B. Analogously a deterministic source branching program (DSP) and a *proper* DSP are defined.

In [KM98] several proper inclusions between complexity classes for OBDDs were proven. We prove that one of the inclusions turns out to be in fact the equality $P = LasVegas$ for OBDDs.

We recall that $ZPP = RP \cap coRP$ where RP is the complexity class determined by "one-sided-error" randomized polynomial computations. The

*Dept. of Computer Science, University of Bonn, and International Computer Science Institute, Berkeley, California. Research partially supported by DFG Grant KA 673/4-1, by the ESPRIT BR Grants 7097, and EC-US 030, and by the Volkswagen-Stiftung. Email: **marek@cs.uni-bonn.de**

†Dept. of Computer Science, University of Bonn. Visiting from Dept. of Theoretical Cybernetics University of Kazan. Research supported by the DAAD-Stiftung and partially by the Russia Fund for Basic Research 96-01-01692. Email: **rustam@ksu.ru**

interesting complexity class is determined by "Las Vegas" (error-free) algorithms, cf. [K98]. For these randomized algorithms the answer "I do not know" is also possible with probability less than $\epsilon < 1/2$. Otherwise it is not allowed to make mistakes: the algorithms give always correct answers. The complexity class *Las Vegas* of functions computable by such polynomially bounded algorithms is equivalent to the complexity class ZPP if no restrictions on number of readings of variables are imposed (cf., e.g., [G77]). It is not evident whether this is true for the complexity classes determined by branching programs with some restrictions on reading inputs. Moreover the following inclusions are straightforward:

$$\begin{aligned} P &\subseteq LasVegas &(1)\\ LasVegas &\subseteq ZPP &(2)\\ ZPP &\subseteq BPP \cap NP \cap coNP. &(3) \end{aligned}$$

In the sequel we consider above complexity classes for read-once branching programs with polynomial sizes.

2 Las Vegas OBDDs

The following function $ADDR(f)_n$ was introduced by S.Jukna ([J88]), and also used by Sauerhoff ([S98b]). We follow the notation of [S98b].

Definition. Let $n = 2^l, l = 2^r, m = n/l = 2^{(l-r)}$. The variables of the function $ADDR(f)_n$ determine $(l \times m)$-matrix. For $i = 0, \ldots, l-1$, $x^i = (x_{im}, \ldots, x_{(i+1)m-1})$ is the i-th row of this matrix. Let $f : \{0,1\}^m \to \{0,1\}$ be an arbitrary function which can be computed by a deterministic read-once branching program in polynomial size. Then $ADDR(f)_n : \{0,1\}^n \to \{0,1\}$ is defined by

$$ADDR(f)_n(x_0, \ldots, x_{n-1}) := x_a, a := |(\lambda(x^0), \ldots, \lambda(x^{l-1}))|_2,$$

where $|(y_0, \ldots, y_{s-1})|_2 := \sum_{i=o}^{r-1} 2^i y_i$ for an arbitrary vector $(y_0, \ldots, y_{s-1}) \in \{0,1\}^s$.

It is shown in [S98b] that the function

$$ADDR(f)_n \in LasVegas{-}BP1 \setminus P{-}BP1$$

and therefore the $LasVegas{-}BP1$ and $P{-}BP1$ complexity classes are different, i.e. the inclusion (1) is proper for BP1s. For OBDDs, we have a surprisingly different result.

Theorem 1

$$P\text{-}OBDD = LasVegas\text{-}OBDD$$

i.e. the inclusion (1) is the equality for OBDDs.

Proof. It is shown in [DHRS97] that the one-way communication complexity of $LasVegas$ computation is at least one half of the one-way deterministic communication complexity. We adopt the proof technique for this result to prove that $P = LasVegas$ for OBDDs. We recall first some ideas of [DHRS97], and adopt them towards our construction.

A *Las Vegas* branching program B computing a function h can be considered as a collection of, say, m deterministic branching programs $B_1, \ldots, B_m$ with assigned probabilities $p_1, \ldots, p_m$. Note that m can depend exponentially on the number of input variables. For any input, B_i may compute results $0, 1$ or 2 (stands for "I do not know"). Since B is a *Las Vegas* OBDD, no OBDD B_i ever errs.

There is a deterministic OBDD B' (with no size restriction) computing the same function h and having the same order as P of reading inputs. Let B' be of the minimum complexity. W.l.g. suppose that OBDDs $B, B_1, \ldots, B_m$ are leveled. For any number l, $1 \leq l < n$, l-th level contains only such nodes for which exactly l variables have been read. These nodes correspond to *maximum* different rows of a communication matrix $C(B')_{l\times(n-l)}$ for OBDD B'. The same is true for OBDDs B_i where communication matrix $C(B_i)_{l\times(n-l)}$ has the same elements as $C(B')_{l\times(n-l)}$ except for some elements that are equal to 2. Following [DHRS97], if $C(B')_{l\times(n-l)}$ has r different rows there is an i, $1 \leq i \leq m$ such that $C(B_i)_{l\times(n-l)}$ has at least $\sqrt{r}$ different rows. Therefore, B_i has at least $\sqrt{r}$ nodes on the l-level.

Because B' is of the minimum complexity, it follows that if B has a polynomial complexity than B' is an OBDD of polynomial size as well. $\square$

Let B be a read-once branching program with the variable set $X = \{x_1, \ldots, x_n\}$, and let (X_1, X_2) be a partition of X. B is called *weakly-ordered with respect to* (X_1, X_2) if all computation paths leading from the source to a sink can be decomposed into two parts, where on the first part only variables from X_1 are tested and on the second part only variables from X_2 (cf. [AK96], and [S98a]).

Corollary 1 *Let h be a function on a set of variables X such that for any partition (X_1, X_2) of X there exists a Las Vegas read-once branching program weakly-ordered with respect to (X_1, X_2) and of polynomial complexity computing h. Then, for any ordering π on X there is a polynomial complexity OBDD respecting π and computing h.*

Indeed, the same idea as in the proof of Theorem 1 holds for the Corollary. If an optimal (i.e. with the minimum complexity) OBDD computing h has an exponential complexity, then it has a level with exponential number of nodes. This level determines a partition (X_1, X_2) such that any weak-ordered Las Vegas-branching programs with respect to (X_1, X_2) computing h has an exponential complexity.

Given now a complexity class Q. The notation $Q-wOBDD$ corresponds to the analogous class of functions computable by some polynomial size weak-ordered branching programs with respect to some partition (X_1, X_2).

Remark 1 *For* $\mathbf{x} = (x_1 \dots x_m)$ *and* $\mathbf{y} = (y_1 \dots y_s)$*, the function* $h(\mathbf{x}, \mathbf{y}) = ADDR(f)_n(\mathbf{x}) \bigvee (\bigvee_{i=1}^{s} y_i)$ *is in* $LasVegas - wOBDD \setminus P - wOBDD$*.*

Indeed, if all y_i are equal to 1, the function h is equal to $ADDR(f)_n$ that is "hard" for deterministic read-once branching programs (cf. [JRSW97], see also [S98b]). Let the $\mathbf{x}$-part of inputs determines the subset X_1 of the set of variables X. Then a polynomial size weak-ordered randomized branching program B with respect to the partition (X_1, X_2) has 2 parts. First part is a Las Vegas read-once branching program *computing* $ADDR(f)_n(\mathbf{x})$ ([S98b]). The second part of B has a source in rejecting sink of the first part and computes $\bigvee_i y_i$. Therefore $h \in LasVegas - wOBDD$.

3 Las Vegas Read-Once Branching Programs

We conjecture that the inclusion 2 is the equality for OBDDs and BP1s. We were however able to prove only the following.

Theorem 2 $ZPP-BP1 \subseteq LasVegas-BP3$*.*

Proof. Let a function h be in $ZPP = RP \cap coRP$ and the randomized branching programs B_1, B_2 $(0, 1-\epsilon)$-compute (with one-sided error) the function h and its negation respectively, $\epsilon < 1/2$. We determine a branching program B in the following way. It starts with the same probability 1/2 the programs B_1 and B_2. The accepting sinks of B_1 and B_2 become the accepting and rejecting sinks of B, respectively. It is evident that on these sinks B gives always correct output for h. We identify other sinks corresponding to the answer "I do not know".The probability of this answer is $1/2(1+\epsilon) < 3/4$. If three copies of B are connected in such a way that the source of a next program is identified with the "I do not know"-sink of a previous one, then the probability of giving the answer "I do not know" for this combination of three programs is less than $(3/4)^3 < 1/2$. □

Theorem 3 *If a function h belongs both to $RP-OBDD$ and to $coRP-OBDD$, and the corresponding OBDDs B_1, B_2 use the same order of deterministic variables and both have a constant (independent of a number of variables of h) number of $RSPs$, then $h \in LasVegas-OBDD$.*

Proof. Firstly we need to prove the following Lemma.

Lemma 1 *If a function h is (a,b)-computed by a polynomial size probabilistic OBDD B with a constant (independent of number of variables of h) number of $RSPs$, then it is (a,b)-computed by a polynomial size probabilistic OBDD B' without a proper RSP.*

Proof. We can obtain from B a polynomial size branching program B'' that (a,b)-computes h, have a constant number of RSPs, for which RSPs C_1, C_2 have common nodes iff C_1 is a RSP of C_2 or vice versa. Analogous property holds for DSPs of B''.

If B'' has proper RSPs without proper RSPs and B'' is a DSP then one can construct a polynomial size RSP without proper RSPs which (a,b)-computes h. This recursive procedure gives a required B'. $\square$

Because of Lemma 1, we can assume that B_1, B_2 do not have a proper RSP. Then we use the "Apply" algorithm (see [B86]) for OBDDs which combines DSPs of B_1 and B_2 with the operator "$\bigwedge$" to compute the graph of deterministic branching programs C_i. By properly randomly choosing C_i we obtain an RSP computing h without an error. Moreover, because for any input, B_1 or B_2 give the correct output, B is a deterministic branching program in a sense that it gives always a correct output (see the Theorem 1), and does not give the answer "I do not know". $\square$

Lemma 3 seems to be interesting because of the following. For any known function computed by OBDDs of different types (probabilistic, nondeterministic, and so on), it holds that all OBDDs use the same order of deterministic variables, and the random or nondeterministic nodes, respectively, are to be tested before deterministic ones.

As for the inclusion (3), we have a conjecture.

Conjecture 1 *For OBDD and BP1, the following is true*

$$RP = BPP \cup NP.$$

If this conjecture is true then $coRP = BPP \cap coNP$, $ZPP = BPP \cap NP \cap coNP$ for OBDD and BP1 (i.e. the inclusion (3) will be the equality for OBDD and BP1).

Acknowledgment

The authors would like to thank Farid Ablayev and Sasha Razborov for helpful discussions on the subject of this paper.

References

[AK96] F. Ablayev and M. Karpinski, *On the power of randomized branching programs*, Proc. ICALP'96, LNCS 1099, Springer, 1996, pp. 348-356.

[B86] R.E.Bryandt, *Graph based algorithms for Boolean function manipulation*, IEEE Trans.Computers, C-35 (8), pp.677-691.

[DHRS97] P.Duris, J.Hromkovič, J.D.P.Rolim and G.Schnitger,*On the Power of Las Vegas for One-Way Communication Complexity, Finite Automata, and Polynomial-time Computations*, ECCC TR97-029, 1997, available at `http://www.ecc.uni-trier.de/eccc/`

[G77] J.Gill, *Computational complexity of probabilistic Turing machines*, SIAM Journal on Computing, 6 (1977), pp.675-695.

[J88] S.Jukna, *Entropy of contact circuits and lower bounds of their complexity*, Theoretical Computer Science, 57: (1988), pp.113-129.

[JRSW97] 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.ecc.uni-trier.de/eccc/`

[K98] M.Karpinski, *On the computational power of randomized branching programs*, Proc. Randomized Algorithms 1998, Brno, 1998, pp.1-12.

[KM98] M.Karpinski and R.Mubarakzjanov, *Some separation problems on randomized OBDDs*, Research Report No. 85196-CS, University Bonn, 1998.

[S98a] M.Sauerhoff, *Randomness and Nondeterminism are Incomparable for Read-Once Branching Programs*, ECCC, TR98-018, 1998, available at `http://www.eccc.uni-trier.de/eccc/`

[S98b] M.Sauerhoff,*Comment 1 on the paper: Randomness and Nondeterminism are Incomparable for Read-Once Branching Programs*, ECCC, TR98-018, 1998, available at `http://www.eccc.uni-trier.de/eccc/`