Electronic Colloquium on Computational Complexity, Report No. 48 (1999)

ASYMPTOTICALLY OPTIMAL BOUNDS FOR OBDDs AND THE SOLUTION OF SOME BASIC OBDD PROBLEMS

Beate Bollig* and Ingo Wegener*

FB Informatik, LS2, Univ. Dortmund, 44221 Dortmund, Germany
bollig,wegener@ls2.cs.uni-dortmund.de

Abstract. Ordered binary decision diagrams (OBDDs) are nowadays the most common dynamic data structure or representation type for Boolean functions. Among the many areas of application are verification, model checking, and computer aided design. For many functions it is easy to estimate the OBDD size but asymptotically optimal bounds are only known in simple situations. In this paper, methods for proving asymptotically optimal bounds are presented and applied to the solution of some basic problems concerning OBDDs. The largest size increase by a synthesis step of π-OBDDs followed by an optimal reordering is determined as well as the largest ratio of the size of deterministic finite automata, quasi-reduced OBDDs, and zero-suppressed BDDs compared to the size of OBDDs. Moreover, the worst case OBDD size of functions with a given number of 1-inputs is investigated.

1 Introduction and results

Branching programs (BPs) are a well established representation type or computation model for Boolean functions. Its size is tightly related to the nonuniform space complexity (see e.g. Wegener (1987)). Hence, one is interested in exponential lower bounds for more and more general types of BPs (for the latest breakthrough for semantic linear depth BPs see Ajtai (1999)). In order to use variants of BPs as dynamic data structure one needs a BP variant such that a list of important operations (see e.g. Wegener (2000)) can be performed efficiently. E.g., for verification, model checking, and a lot of CAD applications we need an efficient test whether a representation has a satisfying input (satisfiability test) and an efficient test whether two representations describe the same function (equality test). These are NP-hard problems for general BPs.

Bryant (1986, 1992) has presented π-OBDDs as a simple BP variant allowing efficient algorithms for all important operations. Although we now have efficient algorithms for more general and, therefore, more compact representation types, π-OBDDs are used in most applications and the use of an OBDD package (Somenzi (1998)) is nowadays a standard technique.

* Supported in part by DFG We 1066/8.

ISSN 1433-8092

Definition 1. *Let $X_n = \{x_1, \dots, x_n\}$ be a set of Boolean variables. A variable ordering π on X_n is a permutation on $\{1, \dots, n\}$ leading to the ordered list $x_{\pi(1)}, \dots, x_{\pi(n)}$ of the variables.*

Definition 2. *A π-OBDD on X_n (see Figure 1) is a directed acyclic graph $G = (V, E)$ whose sinks are labeled by Boolean constants and whose non sink (or inner) nodes are labeled by Boolean variables from X_n. Each inner node has two outgoing edges one labeled by 0 and the other by 1. The edges between inner nodes have to respect the variable ordering π, i.e., if an edge leads from an x_i-node to an x_j-node, $\pi^{-1}(i) \leq \pi^{-1}(j)$ (x_i precedes x_j in $x_{\pi(1)}, \dots, x_{\pi(n)}$). Each node v represents a Boolean function $f_v : \{0,1\}^n \to \{0,1\}$ defined in the following way. In order to evaluate $f_v(a)$, $a \in \{0,1\}^n$, start at v. After reaching an x_i-node choose the outgoing edge with label a_i until a sink is reached. The label of this sink defines $f_v(a)$. The size of the π-OBDD G is equal to the number of its nodes.*

Bryant (1986) has already shown that the minimal-size π-OBDD for a function f is unique (up to isomorphism) and it is called the reduced π-OBDD (or shortly the π-OBDD) for f. Its size is described by the following structure theorem (Sieling and Wegener (1993)). In order to simplify the description we describe the theorem only for the special case where π equals the identity $id(i) = i$.

Theorem 1. *The number of x_i-nodes of the id-OBDD for f is the number s_i of different subfunctions $f_{|x_1=a_1,\dots,x_{i-1}=a_{i-1}}$, $a_1, \dots, a_{i-1} \in \{0,1\}$, essentially depending on x_i (a function g depends essentially on x_i if $g_{|x_i=0} \neq g_{|x_i=1}$).*

It is a simple corollary that the number s_i^* of different subfunctions $f_{|x_1=a_1,\dots,x_{i-1}=a_{i-1}}$, $a_1, \dots, a_{i-1} \in \{0,1\}$, is a lower bound on the *id*-OBDD size of f. Obviously, $\lceil \log s_i^* \rceil$ is the one-way deterministic communication complexity of f if Alice holds $x_1, \dots, x_{i-1}$ and Bob holds $x_i, \dots, x_n$. (See Hromkovič (1997) and Kushilevitz and Nisan (1997) for the theory of communication complexity.) For non-constant functions the *id*-OBDD size of f equals $s_1 + \cdots + s_n + 2$ (2 for the sinks). If many s_i have asymptotically the same and the largest value, one-way communication complexity is not strong enough to obtain asymptotically optimal bounds. Moreover, we have the freedom to choose an appropriate variable ordering for f. Let π-OBDD(f) denote the π-OBDD size of f.

Definition 3. *The OBDD size of f (denoted by OBDD(f)) is the minimum of all π-OBDD(f).*

Using Theorem 1 or one-way communication complexity a lot of exponential lower bounds on the OBDD size of functions have been proved (see e.g. Wegener (2000)). But there are only a few functions whose OBDD size is asymptotically known exactly. These are functions with linear OBDD size, symmetric functions, and a few more functions. The difficulty in proving asymptotically optimal bounds is often the necessity to prove asymptotically optimal bounds for many levels and all variable orderings.

As an example of an exponential but nevertheless unsatisfactory lower bound we mention the best known lower bound on the OBDD size of the middle bit (position $n-1$) of multiplication. The bound (Bryant (1991)) equals $2^{n/8}$. Since the function is defined on $2n$ variables, the trivial upper bound is of size $2^{2n}/n$. People working on OBDDs agree on the conjecture that the OBDD size is at least of order 2^n and millions of OBDD nodes are not sufficient to represent the middle bit of multiplication for $n = 32$ or $n = 64$. The best known lower bound only gives the value 256 for $n = 64$. Hence, the known lower bound does not answer the question on the OBDD size in real life applications (verification of multipliers and dividers). This problem remains open and we only use it as a motivation to develop lower bound methods for asymptotically optimal OBDD bounds. We solve problems motivated from OBDD applications, automata theory, and complexity theory.

In order to use OBDDs we have to transform a logical description of a function, e.g. a circuit, into an OBDD representation. This is done by a sequence of binary synthesis steps. A binary synthesis step computes a π-OBDD G_h for $h = f \otimes g$ ($\otimes$ is a binary Boolean operation like AND or EXOR) from π-OBDDs G_f and G_g for f resp. g. Bryant (1991) has shown how this can be done in time $O(|G_f| \cdot |G_g|)$ and he has presented an example that the result may need size $\Theta(|G_f| \cdot |G_g|)$. His example has two drawbacks. The chosen variable ordering is bad for h, f, and g (and, therefore, such a synthesis step will not occur in applications) and the functions f and g depend essentially on disjoint sets of variables. It is not too hard to present an example without these drawbacks. But this is nevertheless not the answer to the question about the worst case for the binary synthesis problem. If a binary step leads to a π-OBDD much larger than the given π-OBDDs, all recent OBDD packages start to look for a better variable ordering. Although the search for an optimal OBDD variable ordering is NP-hard (Bollig and Wegener (1996)) and this holds even for the corresponding approximation problems for arbitrary constant factors (Sieling (1998)), heuristic algorithms like sifting (Rudell (1993)) often lead to very good results. Hence, the main step is a binary synthesis step followed by reordering. This leads to the problem whether it is possible that OBDD$(h) = \Theta(\pi\text{-OBDD}(f) \cdot \pi\text{-OBDD}(g))$ for functions f and g essentially depending on all considered variables. (Here we should mention the folklore result (see Wegener (2000)) that OBDD(h) may be exponential even if OBDD(f) and OBDD(g) are linear but the linear size only is possible for different variable orderings.) In Section 4, we solve the problem by representing an example where π-OBDD$(f_n) = \Theta(n)$, π-OBDD$(g_n) = \Theta(n)$, and OBDD$(h_n) = \Theta(n^2)$. This surely is the less surprising answer but the lower bound proof for the OBDD size of h_n has some interesting features.

Some applications (Ochi, Yasuoka, and Yajima (1993)) work with a restricted variant of π-OBDDs which may be called leveled π-OBDDs or quasi-π-OBDDs (π-QOBDDs).

Definition 4. *A π-QOBDD is a π-OBDD with the additional property that each edge leaving an $x_{\pi(i)}$-node, $i < n$, reaches an $x_{\pi(i+1)}$-node.*

We are interested in QOBDDs also because of their tight relationship to deterministic finite automata (DFAs) for so-called Boolean languages L where $L \subseteq \{0,1\}^n$ for some n. It is an easy exercise to verify for $L_f = f^{-1}(1)$ that

$$\mathrm{DFA}(L_f) \leq id\text{-QOBDD}(f) \leq \mathrm{DFA}(L_f) + n.$$

Hence, *id*-QOBDDs and DFAs are almost the same. For general regular languages consisting of strings of different lengths it makes no sense to discuss different "variable orderings" or permutations of the input string. For Boolean languages, a π-DFA may apply the reordering π to all inputs of length n. The above inequality can be generalized to

$$\pi\text{-DFA}(L_f) \leq \pi\text{-QOBDD}(f) \leq \pi\text{-DFA}(L_f) + n.$$

Moreover, it is obvious that

$$\pi\text{-QOBDD}(f) \leq (n+1) \cdot \pi\text{-OBDD}(f)$$

and this bound is tight for the constant functions (syntactically depending on n variables). It is also not too difficult to see that $\pi\text{-QOBDD}(f) = \Theta(n \cdot \pi\text{-OBDD}(f))$ for some function f essentially depending on all n variables.

Definition 5. *The multiplexer MUX_n (or direct storage access function DSA_n) is defined on $n+k$ variables $a_{k-1}, \ldots, a_0, x_0, \ldots, x_{n-1}$ where $n = 2^k$. $MUX_n(a,x) = x_{|a|}$ where $|a|$ is the number whose binary representation equals $(a_{k-1}, \ldots, a_0)$.*

Let π be the variable ordering $(a_{k-1}, \ldots, a_0, x_0, \ldots, x_{n-1})$. Then $\pi\text{-OBDD}(\mathrm{MUX}_n) = \mathrm{OBDD}(\mathrm{MUX}_n) = 2n+1$. The π-OBDD (see Figure 1) starts with a complete binary tree on the a-variables. For the path where $|a| = i$ it is sufficient to test x_i. For the π-OBDD we need i extra nodes before the x_i-node and $n-1$ extra nodes before each of the sinks. Hence, $\pi\text{-QOBDD}(\mathrm{MUX}_n) = \frac{1}{2}n^2 + \frac{7}{2}n - 1 = \Theta(n \cdot \pi\text{-OBDD}(\mathrm{MUX}_n))$.

But in order to compare the size of OBDDs and QOBDDs (and DFAs for Boolean languages with reordering) we ask whether $\mathrm{QOBBD}(f_n) = \Theta(n \cdot \mathrm{OBDD}(f_n))$ for functions f_n essentially depending on n variables. This is the question whether the possibility of OBDDs to omit the test of variables may save a size factor of $\Theta(n)$.

Since it is the main rule of thumb for the variable ordering problem to test control or address variables (like the a-variables of MUX_n) before the data variables (like the x-variables of MUX_n), it was a well-established conjecture that the considered variable ordering π is optimal for QOBDDs for MUX_n and that $\mathrm{QOBDD}(\mathrm{MUX}_n) = \Theta(n^2)$. In Section 2, we prove the surprising result that $\mathrm{QOBDD}(\mathrm{MUX}_n) = \Theta(n^2/\log n)$.

Moreover, $\pi\text{-ZBDD}(\mathrm{MUX}_n) = \Theta(n^2)$ and $\mathrm{ZBDD}(\mathrm{MUX}_n) = \Theta(n^2/\log n)$ for zero-suppressed BDDs (ZBDDs) (defined in Section 2) which are used in many applications (see e.g. Minato (1993, 1994)). This is the first example of a function (moreover, a "natural" function) and of BDD models of practical

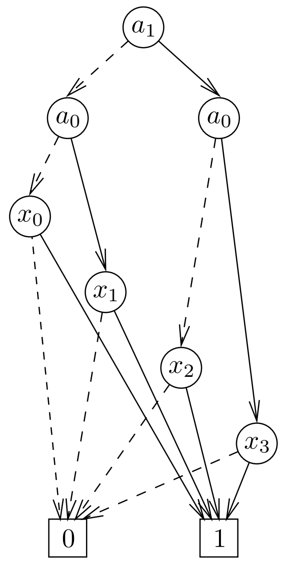

Fig. 1. *An OBDD for MUX_4 (dotted edges are edges with label 0 and solid edges are edges with label 1).*

relevance that the rule of thumb "control variables before data variables" is wrong. In Section 3, we present a function f_n essentially depending on n variables such that $\mathrm{QOBDD}(f_n) = \Theta(n \cdot \mathrm{OBDD}(f_n))$ proving that the freedom of OBDDs to omit tests may indeed decrease the size by a factor of $\Theta(n)$.

In Section 5, we investigate the dependence of the OBDD size on the size of $f_n^{-1}(1)$. Let $N(a(n))$ be the number of Boolean functions f where $|f_n^{-1}(1)| \leq a(n)$. The standard counting argument proves the existence of functions f_n where $|f_n^{-1}(1)| \leq a(n)$ such that its OBDD size and even its circuit size is $\Omega\big(\log N(a(n))/\log\log N(a(n))\big)$. On the other hand, obviously $\mathrm{OBDD}(f_n) \leq \mathrm{O}(na(n))$ for these functions. For $a(n) = 2^n$, the lower bound of size $2^n/n$ is optimal (see Wegener (2000)). For $a(n) = 1$, the upper bound of size n is optimal. The question is how large can we choose $a(n)$ such that we can define functions f_n where $|f_n^{-1}(1)| \leq a(n)$ and $\mathrm{OBDD}(f_n) = \Theta(n\, a(n))$. We describe such functions for polynomially increasing $a(n)$.

2 QOBDDs, DFAs, and ZBDDs for the multiplexer

In this section, we determine the size of QOBDDs, DFAs with reordering, and ZBDDs for the representation of the multiplexer.

Theorem 2. *$QOBDD(MUX_n) = \Theta(n^2/\log n)$.*

This result implies by the discussion in Section 1 the same result for DFAs with reordering.

Proof of Theorem 2. First, we prove for some variable ordering π that the π-QOBDD size of MUX_n is $O(n^2/\log n)$. Let $m := k - \lfloor \log k \rfloor + 1$. The variable ordering π is given by

$$a_{k-1}, \ldots, a_{k-m}, x_0, \ldots, x_{n-1}, a_{k-m-1}, \ldots, a_0.$$

The x-variables are partitioned to 2^m groups such that the indices of the variables of each group agree in their binary representation in the m most significant bits. The size of each group is $n/2^m$. Figure 2 shows a schematic description of the π-QOBDD representing MUX_n. Triangles are complete binary trees. Vertical lines represent tests of x-nodes where both edges leaving a node reach the same successor. We start with a complete binary tree of the first m a-variables. The

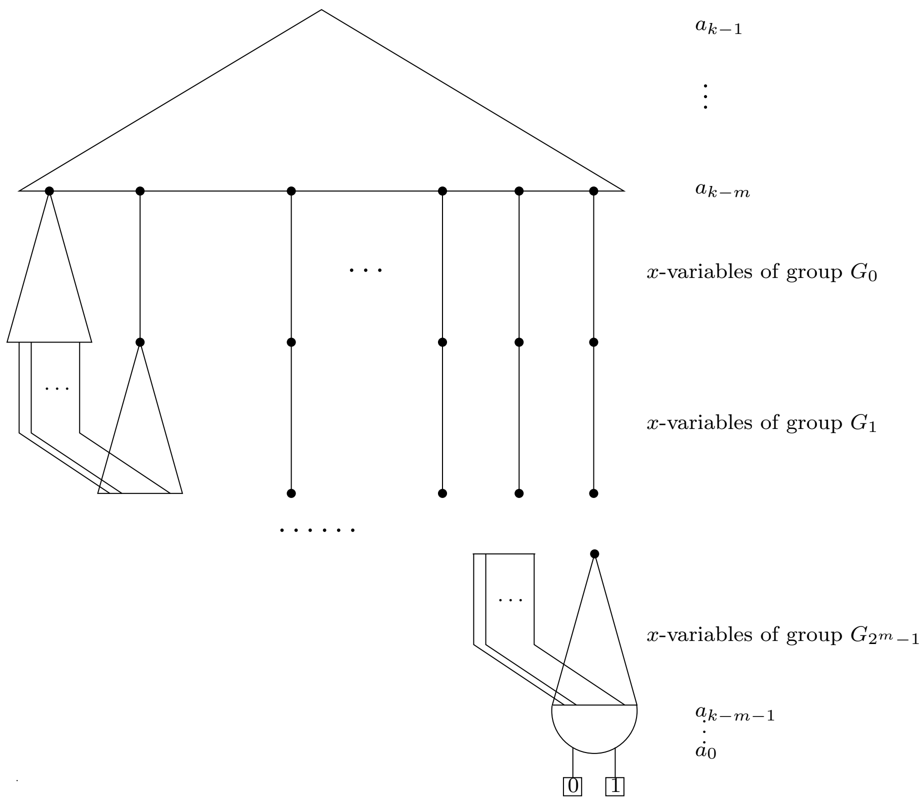

Fig. 2. *A schematic description of the π-QOBDD representing MUX_n.*

tree has 2^m leaves and $2^m - 1$ inner nodes. Then we test the x-variables of group G_0. Only the subfunction where $a_{k-1} = \cdots = a_{k-m} = 0$ essentially depends on these variables. For all other groups we need x_i-nodes, since we consider QOBDDs. Hence, we need one complete binary tree with $2^{n/2^m}$ leaves and $2^{n/2^m} - 1$ inner nodes and $(2^m - 1)n/2^m < n$ further nodes which could be eliminated in OBDDs. One leaf can be replaced by the 0-sink. The same arguments work for the next group. Here the width for the first group is $2^{n/2^m}$ and the total width is bounded by $2 \cdot 2^{n/2^m} + 2^m - 2$. The crucial argument is the following one. We can merge the $2^{n/2^m}$ nodes for the case $(a_{k-1}, \ldots, a_{k-m}) = (0, \ldots, 0, 0)$ with the $2^{n/2^m}$ nodes for the case $(a_{k-1}, \ldots, a_{k-m}) = (0, \ldots, 0, 1)$. We only have to store the data vector namely the x-vector. The result only depends on the further address bits. The width after the tests of the x-variables of group G_{2^m-1} equals $2^{n/2^m} - 1$. The size of the last $k - m$ levels is bounded by $2^{2^{k-m}}$. The total size is bounded above by

$$2^m - 1 + 2^{2^{k-m}} + n(2 \cdot 2^{n/2^m} + 2^m - 2) \le n2^m + 2n2^{n/2^m} + 2^{2^{k-m}}.$$

By the choice of m,

$$2n/\log n \le 2^m \le 4n/\log n$$

and

$$\begin{aligned} n2^m &\le 4n^2/\log n, \\ 2n2^{n/2^m} &\le 2n2^{(\log n)/2} \le 2n^{3/2}, \end{aligned}$$

and

$$2^{2^{k-m}} = 2^{2^{\lfloor \log k \rfloor - 1}} \le 2^{(\log n)/2} \le n^{1/2}.$$

Now we prove the lower bound for arbitrary variable orderings π. It is sufficient to prove a lower bound of size $\Omega(n/\log n)$ on the size of $\Omega(n)$ x-levels of π-QOBDDs.

We claim that the x_i-level of a π-QOBDD representing MUX_n has a size of $\Omega(n/\log n)$ if x_i belongs to the second quarter of all x-variables with respect to π. Let l be the number of address bits tested before x_i. Since the size of the x_i-level does not depend on the ordering of the variables tested before x_i, we assume that π starts with l address variables, w.l.o.g. $a_{k-1}, \ldots, a_{k-l}$. This leads to 2^l groups of x-variables each of size $n/2^l$ (compare the proof of the upper bound). If $r < n/2^l$ variables of a group are tested and x_j is a variable of this group which is not yet tested, all subfunctions for the corresponding assignment to the tested address variables essentially depend on x_j while all other assignments to the tested address variables lead to subfunctions which do not essentially depend on x_j. Hence, nodes for such groups cannot be merged but nodes for groups completely tested can be merged (see the upper bound).

Case 1: $l \ge k - \log k - 2$. Since at most half the x-variables have been tested, at most half of the groups have been tested completely. Therefore, we have at least 2^{l-1} groups each represented by at least one node. This leads to a lower bound of $2^{l-1} \ge \frac{1}{8}n/\log n$.

Case 2: $l < k - \log k - 2$. The number of groups is bounded above by $2^l < \frac{1}{4}n/\log n$. We have already tested $\frac{1}{4}n$ x-variables. Hence, there is one group such that at least $\frac{1}{4}n/(\frac{1}{4}n/\log n) = \log n$ variables of this group have been tested. This group is represented by at least $2^{\log n} = n$ nodes.

Altogether, we have proved that the size of each of at least $n/4$ levels is bounded below by $\frac{1}{8}n/\log n$. □

Definition 6. *A π-ZBDD (zero-suppressed BDD with variable ordering π) G shares its syntax with π-OBDDs. Let v be a node of G and let f_v be the Boolean function represented by G as a π-OBDD. The π-ZBDD G represents at v the following function g_v. If the computation path for f_v and the input a contains x_i-nodes for all variables x_i where $a_i = 1$, then $g_v(a) := f_v(a)$. Otherwise, $g_v(a) := 0$.*

In π-ZBDDs we may omit x_i-nodes representing functions which are called 1-simple with respect to x_i, i.e., functions h such that $h_{|x_i=1} \equiv 0$. It is not possible to omit x_i-nodes representing functions which do not essentially depend on x_i.

Corollary 1. *ZBDD(MUX_n) $= \Theta(n^2/\log n)$.*

Proof. The upper bound follows from Theorem 2, since, by definition, ZBDD$(f) \leq$ QOBDD(f) for all f. The lower bound proof follows the lines of the lower bound proof of Theorem 2 with the following additional remark.

A level of a π-ZBDD can only save the nodes for 1-simple functions compared to π-QOBDDs. The crucial observation is that

$$\mathrm{MUX}_{n|x_i=0} \leq \mathrm{MUX}_{n|x_i=1}$$

for all data variables x_i. This implies that a subfunction of MUX_n which is 1-simple with respect to x_i is the constant 0. Hence, the x_i-level of a π-ZBDD representing MUX_n is at most by 1 smaller than the corresponding level of a π-QOBDD. □

The rule of thumb "control variables before data variables" does not lead to minimal-size QOBDD and ZBDD representations of the practically fundamental multiplexer.

3 The maximal size gap between OBDDs and QOBDDs, DFAs, and ZBDDs

We look for functions f_N essentially depending on all their N variables such that the size gap of OBDDs on one hand and QOBDDs, DFAs with reordering, and ZBDDs on the other hand is a factor of $\Theta(N)$ which is by the discussion in Section 1 the largest possible gap. For such a function it is necessary that many edges in the optimal OBDD omit many tests. Therefore, the multiplexer has been considered as a good candidate for the largest possible gap. But the result of Section 2 implies that the multiplexer only leads to a gap of $\Theta(N/\log N)$. The multiplexer has many more data variables than address variables. We can prove the largest possible gap for a function f_N on $N = n^2+2n$ variables, among them n^2 control or address variables (here called selection variables) $s_0, \ldots, s_{n^2-1}$ and only $2n$ data variables $x_0, y_0, \ldots, x_{n-1}, y_{n-1}$ which lead to n^2 data pairs $x_i y_j$, $0 \le i, j \le n-1$. The data pairs are partitioned to n blocks B_m, $0 \le m \le n-1$, such that B_m contains all pairs $x_i y_{i+m}$ (the indices are computed mod n). We consider the following ordering $p_0, \ldots, p_{n^2-1}$ of the pairs. The pair p_k where $k = in + j$ equals $x_j y_{i+j}$, i.e., we start with the pairs from B_0, followed by the pairs from $B_1, \ldots$. In each block we start with the pair containing x_0, followed by the pair containing x_1, and so on. The main property is that the distance between two pairs containing x_i equals n and the distance between the pairs containing y_i is at least $n-1$. Finally, we define f_N by

$$f_N(s, x, y) = \bigvee_{0 \le k \le n^2-1} s_0 \ldots s_{k-1} \overline{s_k} p_k,$$

i.e., the s-vector selects with its first zero which pair has to be evaluated.

Theorem 3. *OBDD*$(f_N) = \Theta(N)$ *while QOBDD*$(f_N) = \Theta(N^2)$*, DFA*$(f_N) = \Theta(N^2)$*, and ZBDD*$(f_N) = \Theta(N^2)$*.*

Proof. It is obvious that the OBDD size of f_N for the "natural" variable ordering $s_0, \ldots, s_{n^2-1}, x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}$ equals $2n^2 + n + 2 = \Theta(N)$ and that f_N essentially depends on all its variables. This implies OBDD$(f_N) = \Theta(N)$ and the upper bounds for the other models.

In the following we prove the lower bound on the QOBDD size. This implies the lower bound for DFAs with reordering. During the lower bound proof we add remarks that we obtain the same lower bound for ZBDDs.

For the lower bound proof we fix an arbitrary variable ordering π. There are $n^2/24$ levels where a selection variable is tested and the number of already tested selection variables is at least $n^2/8$ and less than $n^2/6$. It is sufficient to prove that each of these levels has a size of $\Omega(n^2)$.

We fix one of the described levels and use the following notation. The sets $T(x)$, $T(y)$, and $T(s)$ contain the x-, y-, and s-variables, resp., which are tested before the considered levels. The sets $R(x)$, $R(y)$, and $R(s)$ contain the corresponding remaining variables. Moreover, $T(p)$ contains the pairs $p_k = x_i y_j$ such that at least one of the variables x_i and y_j is contained in $T(x, y) := T(x) \cup T(y)$ and R(p) contains the remaining pairs. We distinguish whether the size of $T(x, y)$ is small, large, or medium. If it is small, we can argue similarly to the case of the natural variable ordering. If it is large, we have to store too much information on the data variables like in the case of the multiplexer. The most difficult case is the case where $T(x, y)$ is of medium size.

Case 1: $|T(x, y)| \le \frac{1}{9}n$ (small size).
The number of pairs in $T(p)$ can be bounded above by $\frac{1}{9}n^2$. Hence, there are at least $(\frac{1}{8} - \frac{1}{9})n^2 = \Omega(n^2)$ pairs in $R(p)$ such that the corresponding selection variable is in $T(s)$. For each of these pairs $p_k = x_i y_j$ we consider the subfunction g_k of f_N which corresponds to the following assignment to the variables

in $T(x,y,s) := T(x,y) \cup \mathrm{T}(s)$. We assign the value 0 to s_k and all variables in $T(x,y)$ and the value 1 to all variables in $T(s) - \{s_k\}$. The subfunction g_k contains a prime implicant consisting of perhaps some selection variables and p_k. For the subfunction $g_{k'}$, $k' \neq k$, the value 1 is assigned to s_k and the resulting subfunction does not have a prime implicant containing p_k. Hence, we obtain $\Omega(n^2)$ different subfunctions and the size of the level is $\Omega(n^2)$ for π-QOBDDs. The same holds for ZBDDs, since the considered functions are not 1-simple with respect to an s-variable.

Case 2: There are at least $2\log n$ variables in $T(x)$ (or $T(y)$) such that for each of these variables x_i there is a pair $p_k = x_i y_j$ where $s_k \in R(s)$ (large size).

Remark: This case is called "large size", since one of the conditions $|T(x)| \geq \frac{1}{6}n + 2\log n$ and $|T(y)| \geq \frac{1}{6}n + 2\log n$ is sufficient for Case 2. First, we prove this statement. If w.l.o.g. $|T(x)| \geq \frac{1}{6}n + 2\log n$, there are at least $\frac{1}{6}n^2 + 2n\log n$ pairs whose x-variable is contained in $T(x)$. Since, by the choice of the level, $|T(s)| \leq \frac{1}{6}n^2$, there are at least $2n\log n$ pairs $p_k = x_i y_j$ such that $s_k \in R(s)$ and $x_i \in T(x)$. By the pigeonhole principle, we can choose $2\log n$ pairs $p_k = x_i y_j$ with different variables $x_i \in T(x)$ and selection variables $s_k \in R(s)$.

We prove a lower bound of size n^2 for the size of the level by investigating the following n^2 assignments to the variables in $T(x,y,s)$. We assign 1 to all s- and y-variables, 0 to all x-variables which do not belong to a chosen pair, and arbitrary values to all x-variables belonging to a chosen pair. We claim that we obtain n^2 different subfunctions leading to the bound n^2 on the size of this level. Let $p_k = x_i y_j$ be one of the chosen pairs. Let s^* be the conjunction of all s_m, $m < k$, contained in $R(s)$. Let $y^* = y_j$, if $y_j \in R(y)$, and $y^* = 1$, otherwise. The subfunction contains the prime implicant $s^* \overline{s_k} y^*$ iff the value 1 is assigned to x_i. Hence, different assignments lead to different subfunctions. Such a subfunction can only be 1-simple with respect to s_k. This implies a lower bound of n^2 for π-QOBDDs and of $n^2 - 1$ for π-ZBDDs.

Case 3: Not Case 1 or Case 2 (medium size).
Using the Remark the following conditions are fulfilled:

- $|T(x,y)| > \frac{1}{9}n$.
- $|T(x)| < \frac{1}{6}n + 2\log n$ and $|T(y)| < \frac{1}{6}n + 2\log n$.
- There are less than $2\log n$ variables x_i in $T(x)$ (and also less than $2\log n$ variables y_j in $T(y)$) such that $s_k \in R(s)$ for some pair $p_k = x_i y_j$.

The first condition implies that $|T(x)| \geq \frac{1}{18}n$ or $|T(y)| \geq \frac{1}{18}n$. We assume that $|T(x)| \geq \frac{1}{18}n$ (the other case can be handled similarly). The third condition implies the existence of a subset $T'(x)$ of $T(x)$ of size $\frac{1}{18}n - 2\log n \geq \frac{1}{20}n$ (for large n) such that $s_k \in T(s)$ for all pairs $p_k = x_i y_j$ where $x_i \in T'(x)$. The condition $|T(y)| < \frac{1}{6}n + 2\log n \leq \frac{1}{5}n$ (for large n) implies for each $x_i \in T'(x)$ the existence of at least $\frac{4}{5}n$ pairs $p_k = x_i y_j$ such that $s_k \in T(s)$ and $y_j \in R(y)$. Altogether, we argue about these $\frac{4}{5} \cdot \frac{1}{20}n^2 = \frac{1}{25}n^2$ pairs.

We have partitioned the set of all n^2 pairs $p_0, \dots, p_{n^2-1}$ into n blocks $B_0, \dots, B_{n-1}$ such that $B_i = \{p_{in}, \dots, p_{in+n-1}\}$.

Claim 1. *There are $\frac{7}{9}n$ blocks each containing at least $\frac{1}{200}n$ of the chosen pairs.*

Claim 1 is proved by simple counting arguments. First, we mark all $\frac{1}{20}n^2$ pairs $p_k = x_i y_j$ such that $x_i \in T'(x)$. Each block contains $\frac{1}{20}n$ marks because of the chosen ordering of the pairs. Each $y_j \in T(y)$ is combined with each x_i by a pair. We erase the marks for the pairs where $y_j \in T(y)$, altogether at most $\frac{1}{100}n^2$ marks. A block is called good if it still contains $\frac{1}{200}n$ marks, i.e., if at most $\frac{9}{200}n$ marks have been erased. The number of bad blocks is bounded above by $\frac{(1/100)n^2}{(9/200)n} = \frac{2}{9}n$. Hence, we still have $\frac{7}{9}n$ good blocks.

We investigate the set P of the $\Omega(n^2)$ chosen pairs p_k belonging to the $\frac{7}{9}n$ good blocks. For each such pair $p_k = x_i y_j$ we know that $s_k \in T(s)$, $x_i \in T(x)$, and $y_j \in R(y)$. Moreover, we define a subfunction g_k of f_N by assigning the following values to the tested variables. We assign the value 0 to s_k and the value 1 to all other variables in $T(s)$. We assign the value 1 to x_i and the value 0 to all other variables in $T(x, y)$. The function g_k has a prime implicant consisting of y_j and perhaps some s-variables. Hence, it is not 1-simple with respect to an s-variable. It is possible that $g_k = g_l$, $k \neq l$, but then we have some implications on the set $T(s)$.

Claim 2. *If $k < l$ and $g_k = g_l$, then $s_{k+1}, \cdots, s_{l-1} \in T(s)$ and the pairs p_k and p_l contain the same y-variable.*

Claim 2 is proved by contradiction. If p_k contains y_j and p_l contains $y_i \neq y_j$, we obtain different outputs for g_k and g_l by assigning 1 to y_j and all variables in $R(s)$ and by assigning 0 to y_i. If p_k and p_l both contain y_j but $s_m \in R(s)$ for some $k < m < l$, we obtain different outputs for g_k and g_l by assigning 0 to s_m, 1 to all other variables in $R(s)$, 1 to y_j, and 0 to all other variables in $R(x, y)$.

Since each variable is contained in exactly one pair in each block, $g_k = g_l$ implies that p_k and p_l belong to different blocks.

Claim 3. *There are less than $\frac{1}{2}n$ blocks containing some $p_k \in P$ such that $g_k = g_l$ for some $p_l \in P$ and $l \neq k$.*

The proof of Claim 3 is based on the assumption that $|T(s)| < \frac{1}{6}n^2$. Let us assume that $B_{i(1)}, \cdots, B_{i(n/2)}, i(1) < \cdots < i(n/2)$, are blocks where $B_{i(j)}$ contains some pair $p_{i(j)} \in P$ such that $g_{i(j)} = g_l$ for some $p_l \in P$ and $l \neq i(j)$. We know from the construction of the pairs and from Claim 2 that $s_{i(j)}, s_{i(j)+1}, \dots, s_{l-1}, s_l \in T(s)$, if $i(j) < l$, or $s_l, \cdots, s_{i(j)} \in T(s)$, if $i(j) > l$.

The number of these selection variables equals $|l - i(j) + 1| \geq n$, since pairs with the same y-variable have a distance of at least $n - 1$. We claim that $T(s)$ contains at least $\frac{1}{6}n^2$ variables in contradiction to our assumption.

The assumption for $p_{i(j)}$ ensures the existence of n selection variables in $T(s)$ among them $s_{i(1)}$, and $n - 1$ selection variables directly to the left (i.e., $s_{i(1)-1}, \cdots, s_{i(1)-n+1}$) or directly to the right (i.e., $s_{i(1)+1}, \cdots, s_{i(1)+n-1}$) of $s_{i(1)}$. The pair $p_{i(2)}$ may not ensure the existence of further variables in $T(s)$. If $p_{i(1)}$ is at the right border of its block and $p_{i(3)}$ at the left border of its block, also

$p_{i(3)}$ does not ensure the existence of many further variables in $T(s)$. The pair $p_{i(4)}$ ensures that $s_{i(4)}$ and $n-1$ variables directly to the left or to the right are in $T(s)$. These variables are different from the variables counted for $p_{i(1)}$, since $p_{i(1)}$ and $p_{i(4)}$ are separated by 2 blocks of length n each. This argument holds for each third pair proving that $|T(s)| \geq \frac{1}{6}n^2$.
From Claim 1 and Claim 3 we conclude that there are at least $(\frac{7}{9} - \frac{1}{2})n = \frac{5}{18}n$ blocks each containing $\frac{1}{200}n$ pairs such that all the corresponding $\Omega(n^2)$ subfunctions g_k are different. This implies the lower bound $\Omega(n^2)$ on the size of the level. □

4 The maximal size increase of a π-OBDD synthesis step with optimal reordering

In this section, we prove that the synthesis of π-OBDDs essentially depending on the same set of variables can lead to a multiplicative size increase (like the well-known result for DFAs) and that this result even holds if the synthesis can be followed by an optimal reordering.

The functions are defined on $n + 2k$ variables $x_0, \ldots, x_{k-1}, y_0, \ldots, y_{k-1}, z_0, \ldots, z_{n-1}$ where $n = 2^k$. Let $f_n(x, y, z) = \mathrm{MUX}_n(x, z)$ and $g_n(x, y, z) = \mathrm{MUX}_n(y, z)$. These functions do not depend essentially on all variables. Nevertheless, we first investigate f_n, g_n, and $h_n = f_n \oplus g_n$. Afterwards, we define modified functions f_n^*, g_n^*, and $h_n^* = f_n^* \oplus g_n^*$ depending essentially on all variables and having similar properties. We also will argue that we essentially get the same results if we replace $\oplus$ by $+$. The function h_n is defined by

$$h_n(x, y, z) = \bigvee_{0 \leq i,j \leq n-1} (|x| = i) \wedge (|y| = j) \wedge (z_i \oplus z_j).$$

Theorem 4. *Let π^* be the variable ordering $x_0, \ldots, x_{k-1}, y_0, \ldots, y_{k-1}, x_0, \ldots, x_{n-1}$. Then π^*-OBDD$(f_n) = \pi^*$-OBDD$(g_n) = 2n + 1$ but OBDD$(h_n) = \Omega(n^2)$, i.e., a synthesis step followed by optimal reordering can lead to a multiplicative size increase.*

We will see that the effect of the choice of the variable ordering for h_n has interesting features. We have already seen in Section 2 that the representation size of multiplexer based functions has surprising effects. Here we prove a lower bound of size $\Omega(n^2)$ by proving that each of $\Omega(n)$ levels has size $\Omega(n)$. The interesting aspect is that there is not necessarily a block of $\Omega(n)$ levels each of size $\Omega(n)$. It may happen that small levels and large levels occur in a rather irregular order. Nevertheless, we are able to bound the number of small levels in a sufficient way. This is the first proof of an asymptotically optimal OBDD lower bound in such a situation.

Proof of Theorem 4. The upper bounds are obvious. For the lower bound proof we fix an arbitrary variable ordering π.

First, we visualize the situation after the test of some variables. We do not use the communication matrix, since we believe that a different representation better supports the counting of different subfunctions essentially depending on some specific z-variable. Again we use the notation $T(x)$, $T(y)$, and $T(z)$ for the sets of already tested variables. Let $r := T(x)$ and $c := T(y)$. Then we have 2^r partial x-addresses which partition the set of all z-variables into 2^r blocks of size $n2^{-r}$ each. Two z-variables z_i and z_j belong to the same block if the binary representations of i and j agree in the positions belonging to variables in $T(x)$. In the same way we obtain 2^c blocks of size $n2^{-c}$ each corresponding to the variables in $T(y)$. We consider the following $n \times n$-matrix. The rows correspond to the z-variables and they are ordered blockwise with respect to the 2^r row blocks. In each block we order the variables according to the canonical ordering with respect to the vector describing the value of the x-variables which have not been tested yet. The columns also correspond to the z-variables and they are ordered blockwise with respect to the 2^c column blocks. The entry at the z_i-row and the z_j-column equals $z_i \oplus z_j$. Our aim is to prove a lower bound on the size of the z_i-

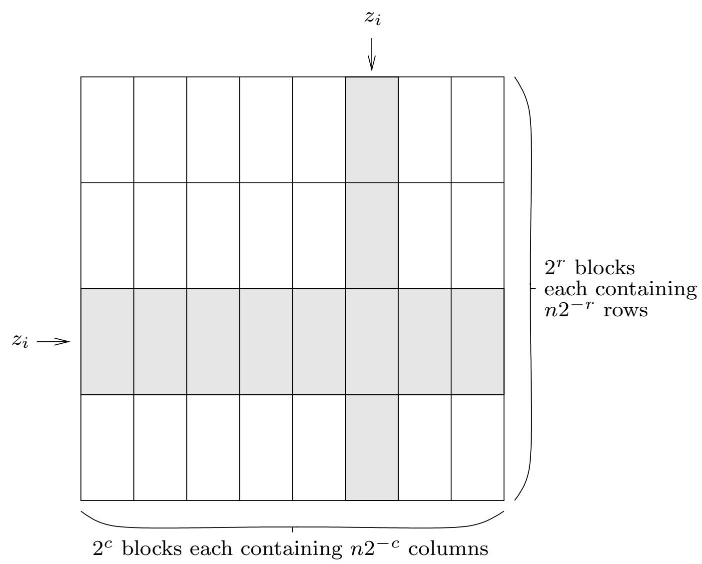

Fig. 3. *A macroscopic view.*

level of the π-OBDD representing h_n. For this reason it is sufficient to investigate those $2^r + 2^c - 1$ assignments to the variables in $T(x, y)$ where at least one of the partial addresses allows the value i. In Figure 3 the corresponding blocks are shaded. Our aim is to count the number of different subfunctions essentially depending on z_i. First, we consider the subfunctions for some fixed assignment to the variables in $T(x, y)$. This leads to a submatrix of the matrix considered above (see Figure 4). In our example the submatrix contains the z_i-row but not the z_i-column. It is called a z_i-row-rectangle. Variables $z_j \in T(z)$ are replaced by a_j. This z_i-row-rectangle is a description of the considered subfunction of h_n. Let s be the number of different variables from $T(z)$ which correspond to a

$$
\begin{array}{c|ccccc|}
 & a_j & \cdots & a_l & \cdots & z_m \\ \hline
a_p & a_p \oplus a_j & \cdots & a_p \oplus a_l & \cdots & a_p \oplus z_m \\
\vdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
a_q & a_q \oplus a_j & \cdots & a_q \oplus a_l & \cdots & a_q \oplus z_m \\
\vdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
z_i & z_i \oplus a_j & \cdots & z_i \oplus a_l & \cdots & z_i \oplus z_m \\
\vdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
z_t & z_t \oplus a_j & \cdots & z_t \oplus a_l & \cdots & z_t \oplus z_m \\ \hline
\end{array}
$$

Fig. 4. *A microscopic view.*

column or row of the considered submatrix. Then we obtain exactly 2^s different subfunctions all essentially depending on z_i. The z_i-row has entries from $\{z_i, \overline{z}_i\}$ in the columns corresponding to variables from $T(z)$. Reading the z_i-row we can compute the assignment to the z-variables corresponding to the columns. Now let us consider a row belonging to a variable from $T(z)$ which does not occur as a column of this block, e.g., the a_p-row. Since we have computed a_j, we can compute a_p from $a_p \oplus a_j$ or from $a_p \oplus z_m$. Similar results hold for z_i-column-rectangles.

We have computed the number of subfunctions essentially depending on z_i for a fixed assignment to the variables from $T(x, y)$ and all assignments to the variables from $T(z)$. Is it possible to obtain the same subfunction for different assignments to the variables from $T(x, y)$? This happens iff the a-entries are replaced in such a way by constants that two z_i-rectangles are equal. We assume that $r < k$ or $c < k$ (if $r = c = k$, all address variables have been tested which leads to an easy subcase). A z_i-row-rectangle R_i differs from a z_i-column-rectangle C_i, since R_i contains entries essentially depending on z_i exactly in the z_i-row while this happens in C_i exactly in the z_i-column. Now we consider w.l.o.g. two z_i-row-rectangles R_i and R'_i. If R_i contains a z_m-column, it contains a column where all entries essentially depend on z_m. This cannot happen in R'_i where at most one row can depend essentially on z_m. Hence, two different z_i-row-rectangles agree iff all z-variables corresponding to the columns have already been tested and the corresponding vectors are equal (remember the agreement that the variables of each block are ordered according to the canonical ordering with respect to the vector describing the value of the y-variables not from $T(y)$). The only-if-part follows from the consideration of the case $|x| = i$. The if-part follows, since the remaining assignments to x-variables and z-variables belonging to rows of the block have the same influence on R_i and R'_i.

Summarizing we can conclude that we are able to determine the size of each z-level of a π-OBDD representing h_n. We still have to prove that $\Omega(n)$ z-levels have size $\Omega(n)$. The first and last z-levels can be very small. We concentrate on the levels where the z-variables at the positions $n/2+1,\dots,3n/4$ of the variable ordering on the z-variables are tested.

Case 1. There is no block such that all corresponding row variables have already been tested. (The same arguments work for the column variables.)

We consider the z_i-column-rectangles. Let r_j be the number of $T(z)$-variables belonging to the jth row block, $1 \le j \le 2^r$. The sum of all r_j is at least $n/2$ (by the choice of the considered levels) and our lower bound arguments lead to the lower bound

$$\sum_{1\le j\le 2^r} 2^{r_j}$$

on the size of the considered z_i-level. This lower bound is minimal if the sum of all r_j is equal to $n/2$ and all r_j are equal to $\frac{n}{2}/2^r$. This leads to the lower bound

$$2^r 2^{n/2^{r+1}}.$$

As long as $n/2^{r+1} \ge 2$, this exponent decreases at least by 1 if r is increased by 1. In these cases the lower bound is decreasing with r. This happens as long as $r+2 \le \log n$. Hence, we obtain an $\Omega(n)$ bound on the size of the z_i-level.

Case 2. Not Case 1 and $r \le \log n - \log\log n$ (or $c \le \log n - \log\log n$).

There is a z_i-column-rectangle where all $n2^{-r}$ z-variables corresponding to the rows of the rectangle have already been tested. This leads to a lower bound of size $2^{n2^{-r}} \ge n$.

Case 3. $r \ge \log n - c^*$ (or $c \ge \log n - c^*$) for some constant c^* ($c^* = 8$ is appropriate in this proof).

We have 2^r z_i-column-rectangles and at least $n/4$ z-variables which have not been tested. Each row block contains $n2^{-r}$ z-variables. Hence, there are at least $\frac{1}{4}2^r = \Omega(n)$ z_i-column-rectangles such that at least one row variable has not been tested.

For each of these rectangles we obtain a lower bound of size 1 and we have shown above that the different z_i-column-rectangles cannot agree and describe different subfunctions. Altogether, we obtain also in this case a lower bound of size $\Omega(n)$.

We obtain the proposed $\Omega(n^2)$ lower bound if $\Omega(n)$ of the $n/4$ considered levels belong to one of the three cases. Hence, we only have to consider the situation where $n/4 - o(n)$ of the considered levels do not fulfil the assumptions of one of the three cases. We assume w.l.o.g. that on $n/8 - o(n)$ of these levels the condition $c \le r$ holds.

On these levels, $\log n - \log\log n < c \le r < \log n - c^*$. Only to simplify the notation we assume that $N = \log\log n$ is an integer. We consider the levels where $r = \log n - \log\log n + t$ has a fixed value. In particular, $1 \le t \le \log\log n - c^*$. We have $2^r = 2^t n/\log n$ z_i-column-rectangles. Since at least $n/4$ z-variables have not been tested, there are at least 2^{r-2} z_i-column-rectangles such that some corresponding row variable has not been tested. Let m be the number of already tested z-variables belonging to the column block containing z_i. This leads to the lower bound $\frac{1}{4}2^r 2^m$. If $m \geq \log\log n - t$, the lower bound is of size $\Omega(n)$. We have 2^c column blocks of size $n2^{-c}$ each. Hence, there are at most $2^c(\log\log n - t) \leq 2^r(\log\log n - t)$ z-levels where we have not proved an $\Omega(n)$ bound.

We estimate the number of bad levels, i.e., those levels where we have not proved an $\Omega(n)$ bound. Let $n \geq 4$. Then the number of bad levels can be estimated by

$$\sum_{1 \leq t < N - c^*} 2^t \frac{n}{\log n}(N-t) =$$
$$\frac{n}{\log n}\left(N(2^{N-c^*}-2) - (N-c^*-1)2^{N-c^*} + 2^{N-c^*} - 2\right) \leq$$
$$\frac{n}{\log n}(c^*+2)2^{-c^*}2^N = (c^*+2)2^{-c^*}n.$$

For $c^* = 8$ these are at most $\frac{10}{256}n$ bad levels out of $\frac{1}{8}n - o(n)$ levels. Hence, also in this situation we have proved the existence of $\Omega(n)$ levels whose size is $\Omega(n)$. This implies the proposed $\Omega(n^2)$ bound. □

If $h'_n = f_n + g_n$, we can use almost the same arguments. If all column variables of a z_i-row-rectangle have been replaced by ones, the corresponding subfunction does not essentially depend on z_i. It is obvious that the lower bound decreases at most by a factor of 2.

In a last step, we generalize our results to functions essentially depending on all their variables. We work with two additional variables namely x_k and y_k. Let $x = (x_k, \ldots, x_0)$, $x^* = (x_{k-1}, \ldots, x_0)$ and let the predicate $(\oplus x = 1)$ be true iff $x_0 \oplus \cdots \oplus x_k = 1$, similarly for y, y^*, and $\oplus y$. Let

$$f^*_n(x,y,z) = \bigvee_{0 \leq i \leq n-1} (|x^*| = i) \wedge (\oplus x = 1) \wedge (\oplus y = 1) \wedge z_i$$

and

$$g^*_n(x,y,z) = \bigvee_{0 \leq j \leq n-1} (|y^*| = j) \wedge (\oplus x = 1) \wedge (\oplus y = 1) \wedge z_j.$$

It is obvious that f^*_n and g^*_n essentially depend on all their variables. For a small representation of f^*_n and g^*_n it is not sufficient to test the x- and y-variables before the z-variables. We choose the variable ordering

$$x_0, y_0, x_1, y_1, \ldots, x_k, y_k, z_0, \ldots, z_{n-1}.$$

Then the OBDD size of f^*_n is bounded above by $11n$. The number of x_i-nodes, $1 \leq i \leq k-1$, equals 2^{i+1}, since we store all tested x-variables and the parity of the tested y-variables. The number of y_i-nodes is twice the number of x_i-nodes. Hence, the first $2k$ levels have a size which is bounded above by $6n-2$. We have to store $|x|$ and the parity of $y_0 \ldots, y_{k-1}$. Hence, we have $2n$ x_k-nodes and $2n$ y_k-nodes followed by n z-nodes and two sinks. The OBDD size of g^*_n and the chosen variable ordering is even bounded above by $8n$. The function $h_n^* = f_n^* \oplus g_n^*$ is described by

$$h_n^*(x,y,z) = \bigvee_{0 \le i,j \le n-1} (|x^*| = i) \wedge (|y^*| = j) \wedge (\oplus x = 1) \wedge (\oplus y = 1) \wedge (z_i \oplus z_j).$$

The same arguments as for h_n lead to an $\Omega(n^2)$ bound on the OBDD size of h_n^* for arbitrary variable orderings. Also here we can replace $\oplus$ by $+$.

5 On the maximal OBDD size with respect to the number of 1-inputs

For the construction of a function $f_{n,k}$ with $|f_n^{-1}(1)| = \binom{n}{k}$ and $\text{OBDD}(f_{n,k}) = \Theta(n|f_n^{-1}(1)|)$ we use the construction of Kovari, Sós, and Turán (1954) for the solution of the well-known problem of Zarankiewicz. Their result can be explained as follows. Let $n = p^2$ for some odd prime p. Let $U := \{0, \dots, n-1\}$ be the universe which is partitioned to p blocks $B_0, \dots, B_{p-1}$ where $B_i = \{ip, \dots, ip+p-1\}$. Then it is possible to define explicitly sets $A_0, \dots, A_{n-1}$ with the following properties:

- $|A_i| = p$ for all i,
- $|A_i \cap A_j| \le 1$ for all $i \ne j$,
- $|A_i \cap B_j| = 1$ for all i and j,
- for all $i \in B_k$ and $j \in B_l$ where $k \ne l$ there exists some m such that $i, j \in A_m$.

For the definition of $f_{n,k}$ we consider for each choice of $0 \le i_1 < i_2 < \dots < i_k \le n$ the set $A_{i_1,\dots,i_k}$ defined as union of all A_{i_j}, $1 \le j \le k$, and the corresponding minterm $m_{i_1,\dots,i_k}$ on $\{x_0, \dots, x_{n-1}\}$ which computes 1 iff $x_i = 1$ exactly for all $i \in A_{i_1,\dots,i_k}$. The function $f_{n,k}$ computes 1 iff one of the minterms $m_{i_1,\dots,i_k}$ computes 1.

Theorem 5. *Let k be a constant. Then $|f_n^{-1}(1)| \le \binom{n}{k}$ and $OBDD(f_{n,k}) = \Theta(n\binom{n}{k})$.*

Proof. By definition, $|f_{n,k}^{-1}(1)| \le \binom{n}{k}$. (For large n, even $|f_{n,k}^{-1}(1)| = \binom{n}{k}$.) This implies the upper bound $n\binom{n}{k} + 2$ on the OBDD size of $f_{n,k}$, since at most $\binom{n}{k}$ of all assignments to some set of variables can be different from the constant 0.

For the lower bound proof we fix a variable ordering π and investigate the set P of all 1-paths namely all computation paths $p_{i_1,\dots,i_k}$ corresponding to the minterms $m_{i_1,\dots,i_k}$. The proof strategy is the following one. We identify a set $P' \subseteq P$ such that two different paths from P' have been split before or at level $\frac{1}{3}n$, i.e., there is a node where one path chooses the 0-edge and the other one chooses the 1-edge. Afterwards, we identify a subset of P' such that two paths from this subset cannot share a node on the levels $\frac{1}{3}n, \dots, \frac{2}{5}n$. We ensure that the size of this subset is $\Omega(N)$ for $N = \binom{n}{k}$. This proves the lower bound.

First we remark that $|A_{i_1,\dots,i_k} \cap A_i| \le k$, if $i \notin \{i_1, \dots, i_k\}$. This has the following consequences. Since (for large n) $p \ge k+2$, the inputs from $f_{n,k}^{-1}(1)$ have a Hamming distance of at least 2 and each 1-path contains n inner nodes. Moreover, an input a' which is the characteristic vector of A' such that $|A' \cap A_i| \ge k+1$ and $A_i \not\subseteq A'$ has the property that $f_{n,k}(a') = 0$.

As next step, we prove that many 1-paths split early. Let I contain the indices of the first $\frac{1}{3}n$ variables according to π. The average size of all $B_i \cap I$ equals $\frac{1}{3}p$ and (for large n) there are two different blocks B_i and B_j such that $|B_i \cap I| \ge \frac{1}{4}p$ and $|B_j \cap I| \ge \frac{1}{4}p$. There are at least $\binom{p/4}{k}\binom{p/4}{k} = \Omega(N)$ (remember that k is a constant) possibilities to choose k elements $i_1, \dots, i_k$ from $B_i \cap I$ and k elements $j_1, \dots, j_k$ from $B_j \cap I$. We identify each such choice with a unique minterm. The pair (i_r, j_r) determines by the properties of the A-sets uniquely a set A_{m_r} such that $i_r, j_r \in A_{m_r}$. Since $A_{m_1,\dots,m_k} \cap B_i = \{i_1, \dots, i_k\}$ and $A_{m_1,\dots,m_k} \cap B_j = \{j_1, \dots, j_k\}$, different choices lead to different 1-paths. Let P' be the set of the chosen $\Omega(N)$ 1-paths. Let us consider two different of these 1-paths or minterms. They correspond to the choices $i_1, \dots, i_k, j_1, \dots, j_k$ and $i'_1, \dots, i'_k, j'_1, \dots, j'_k$ and w.l.o.g. $i_1 \notin \{i'_1, \dots, i'_k\}$. The variable x_{i_1} is tested on one of the first $\frac{1}{3}n$ levels and the first minterm chooses a 1-edge on this level and the second one a 0-edge. Hence, the paths from P' split before or at level $\frac{1}{3}n$.

As final step, we prove that many 1-paths from P' do not merge again before level $\frac{2}{5}n$. Let I^* contain the indices of the first $\frac{2}{5}n$ variables according to π. Let r be the number of rich sets A_i, i.e., sets where $|A_i \cap I^*| \ge p-k$. We prove by contradiction that $r \le p-k-1$. We assume that $|A_{i_1} \cap I^*| \ge p-k, \dots, |A_{i_{p-k}} \cap I^*| \ge p-k$. Since $|A_{i_1} \cap A_{i_2}| \le 1$, $|(A_{i_1} \cup A_{i_2}) \cap I^*| \ge (p-k)+(p-k-1)$. In the same way, we conclude (for large n) that

$$|(A_{i_1} \cup \dots \cup A_{i_{p-k}}) \cap I^*| \ge (p-k) + (p-k-1) + \dots + 1 \ge \frac{1}{2}(p-k)^2 > \frac{2}{5}n$$

in contradiction to $|I^*| = \frac{2}{5}n$.

Let $P'' \subseteq P$ be the set of 1-paths corresponding to sets $A_{i_1,\dots,i_k}$ such that all A_{i_r} are poor, i.e., not rich. The number of rich A-sets has been shown to be at most $p-k-1$. Hence, we have more than $n-p$ poor sets and more than $\binom{n-p}{k}$ sets $A_{i_1,\dots,i_k}$ consisting of poor sets only. Since $n-p = n-o(n)$ and k is a constant, $|P''| > \binom{n-p}{k} = N - o(N)$. Hence, P' and P'' are subsets of P where $|P| = N$, $|P'| = \Omega(N)$, and $|P''| = N - o(N)$. Hence, $|P' \cap P''| = \Omega(N)$. We consider two different paths p_1 and p_2 from $P' \cap P''$. They have split before or at level $\frac{1}{3}n$. We assume w.l.o.g. that p_1 and p_2 correspond to $A_{i_1,\dots,i_k}$ and $A_{j_1,\dots,j_k}$ and split on the x_i-level, $i \in I$, where $i \in A_{i_1,\dots,i_k}$ (w.l.o.g. $i \in A_{i_1}$) and $i \notin A_{j_1,\dots,j_k}$. Now we assume that p_1 and p_2 share the node v on one of the levels between $\frac{1}{3}n$ and $\frac{2}{5}n$. Then the path p^* following p_2 from the source to v and p_1 from v also is a 1-path corresponding to an input a' which is the characteristic vector of some set A'. Since A_{i_1} is poor, at least $k+1$ variables

$x_r, r \in A_{i_1}$, are tested positively on p^*, namely on that part of p_1 which starts at v. Hence, $|A' \cap A_{i_1}| \geq k+1$. Since x_i is tested negatively on p^*, namely on that part of p_2 which stops at v, $A_{i_1} \not\subseteq A'$ and $f_{n,k}(a') = 0$ (as shown above) in contradiction to the construction of p^* as 1-path. This proves Theorem 5. $\Box$

The bounds of Theorem 5 even hold for nondeterministic OBDDs which we do not define explicitly here (see Wegener (2000)). This is obvious for the upper bound. For the lower bound it is sufficient to choose for each 1-input one computation path leading to the 1-sink. Then we may argue in the same way with this set of paths.

Many lower bounds for variants of BPs have been obtained for the characteristic sets of certain linear codes (see e.g. Okol'nishnikova (1993), Jukna (1995), and Jukna and Razborov (1998)). The important property of linear codes is the large Hamming distance between different code words. This seems to be not sufficient to obtain lower bounds on the OBDD size of size $\Omega(n|f^{-1}(1)|)$. Our construction uses the fact that the 1-inputs of our function have a large Hamming distance and additionally the other structural properties of the A-sets.

References

1. Ajtai, M. (1999). A non-linear time lower bound for Boolean branching programs. 40. FOCS, 60–70.
2. Bollig, B. and Wegener, I. (1996). Improving the variable ordering of OBDDs is NP-complete. IEEE Trans. on Computers 45(9), 993–1002.
3. Bryant, R. E. (1986). Graph-based algorithms for Boolean manipulation. IEEE Trans. on Computers 35, 677–691.
4. Bryant, R. E. (1991). On the complexity of VLSI implementations and graph representations of Boolean functions with application to integer multiplication. IEEE Trans. on Computers 40, 205–213.
5. Bryant, R. E. (1992). Symbolic Boolean manipulation with ordered binary decision diagrams. ACM Computing Surveys 24, 535–541.
6. Hromkovič, J. (1997). *Communication Complexity and Parallel Computing.* Springer.
7. Jukna, S. (1995). A note on read-k-times branching programs. RAIRO-Theoretical Informatics and Applications 29, 75-83.
8. Jukna, S. and Razborov, A. (1998). Neither reading few bits twice nor reading illegally helps much. Discrete Applied Mathematics 85, 223–238.
9. Kovari, T. Sós, V., and Turán, P. (1954). On a problem of K. Zarankiewicz. Colloquium Mathematicum 3, 50–57.
10. Kushilevitz, E. and Nisan, N. (1997). *Communication Complexity.* Cambridge University Press.
11. Minato, S. (1993). Zero-suppressed BDDs for set manipulation in combinatorial problems. 30. DAC, 272–277.
12. Minato, S. (1994). Calculation of unate cube set algebra using zero-suppressed BDDs. 31. DAC, 420–424.
13. Ochi, H., Yasuoka, K., and Yajima, S. (1993). Breadth-first manipulation of very large binary-decision diagrams. ICCAD '93, 48–55.

14. Okol'nishnikova, E. A. (1993). On lower bounds for branching programs. Siberian Advances in Mathematics 3(1), 152–166.
15. Rudell, R. (1993). Dynamic variable ordering for ordered binary decision diagrams. ICCAD '93, 42–47.
16. Sieling, D. (1998). On the existence of polynomial time approximation schemes for OBDD minimization. STACS '98, LNCS 1373, 205–215.
17. Sieling, D. and Wegener, I. (1993). NC-algorithms for operations on binary decision diagrams. Parallel Processing Letters 48, 139–144
18. Somenzi, F. (1998). CUDD: CU decision diagram package release 2.3.0. Techn. Rep. of the University of Colorado at Boulder.
19. Wegener, I. (1987). *The Complexity of Boolean Functions*. Wiley-Teubner.
20. Wegener, I. (2000). *Branching Programs and Binary Decision Diagrams - Theory and Applications*. SIAM Monographs in Discrete Mathematics and Its Applications. In print.

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'