Variable Orderings and the Size of OBDDs for Random Partially Symmetric Boolean Functions

Detlef Sieling
FB Informatik, LS II, University of Dortmund, 44221 Dortmund, Federal Republic of Germany

Received 29 October 1996; accepted 24 February 1998

ABSTRACT: The size of ordered binary decision diagrams (OBDDs) strongly depends on the chosen variable ordering. It is an obvious heuristic to use symmetric variable orderings, i.e., variable orderings where symmetric variables are arranged adjacently. In order to evaluate this heuristic, methods for estimating the OBDD size for random partially symmetric functions are presented. Characterizations of cases where, with high probability, only symmetric variable orderings and, with high probability, only nonsymmetric variable orderings lead to minimum OBDD size are obtained. For this analysis estimates for the number of different blocks of random Boolean matrices are used. © 1998 John Wiley & Sons, Inc. Random Struct. Alg., 13, 49–70, 1998

Key Words: ordered binary decision diagrams; partially symmetric functions; symmetric variable orderings; random functions; random Boolean matrices

1. INTRODUCTION

Ordered binary decision diagrams (OBDDs) are the most successful data structure for Boolean functions. They are applied in programs for, e.g., circuit verification, test pattern generation, logic synthesis, model checking, and even in combinatorial applications like counting the number of knight's tours on a chessboard [9]. The

Contract grant sponsor: DFG
Contract grant numbers: We 1066/7 and We 1066/8

reason for the success of OBDDs is the efficient algorithms for the manipulation of Boolean functions represented by OBDDs, which were presented in the seminal paper by Bryant [2]. However, the size of the data structures affects their applicability. The problem of OBDDs is that they are often very large. An important parameter that influences the size of OBDDs is the variable ordering. There are examples of functions with linear OBDD size for some variable orderings and exponential OBDD size for other variable orderings [2]. Hence, algorithms for computing good variable orderings are needed. Unfortunately the computation of optimal variable orderings is known to be *NP*-hard [1] and even hard to approximate [16]. For this reason many heuristics for computing good variable orderings have been proposed (see, e.g., [5, 10, 4]). An important principle of many of these heuristics is that variables belonging together in some sense are placed together in the variable ordering.

The main issue of this paper is the examination of variable orderings and the OBDD size for a special class of functions, namely, partially symmetric functions. A function is called partially symmetric with respect to the partition $V_1, \dots, V_d$ of the set of variables if the function does not change when permuting the variables in each of the sets $V_1, \dots, V_d$. We call these sets symmetry sets. Obviously the following definition of partially symmetric functions is equivalent: The value of the function only depends on the number of 1s assigned to the variables in each symmetry set, but does not depend on the position of these 1s.

Partially symmetric functions are considered in several papers because knowledge of the symmetry sets may help to improve algorithms, e.g., one may obtain better results in the logic synthesis [7]. OBDD-based algorithms for computation of the symmetry sets were presented by Möller, Mohnke, and Weber [11] and Panda, Somenzi, and Plessier [14]. Totally symmetric functions are the special case of partially symmetric functions with only one symmetry set containing all variables. The size and the structure of OBDDs for totally symmetric functions were investigated by Wegener [19].

It is obvious that variables of the same symmetry set belong together. This leads to the following heuristic for the OBDD variable ordering problem: Variables of the same symmetry set should be placed adjacently in the variable ordering. We call such variable orderings symmetric variable orderings. The heuristic to use symmetric variable orderings is supported by experiments of Jeong, Kim, and Somenzi [6]. They computed optimal variable orderings for a set of benchmark circuits and observed that these optimal variable orderings were always symmetric. This leads to the conjecture that for each function, there is some optimal variable ordering that is symmetric. Based on this conjecture, Panda, Somenzi, and Plessier [14] and Panda and Somenzi [13] improved the (heuristic) sifting algorithm due to Rudell [15] for minimizing OBDDs. They integrated the detection of symmetries into the sifting process and glued variables of the same symmetry set together (also for some extended notion of symmetry). The conjecture was disproved by Möller, Molitor, and Drechsler [12], who found an example with the minimum OBDD size 7 for symmetric variable orderings and the minimum OBDD size 6 for arbitrary variable orderings. This example motivates several questions.

The first question is why nonsymmetric variable orderings may admit smaller OBDDs. By a simple example we show that nonsymmetric variable orderings may allow mergings of isomorphic parts of an OBDD that are not possible with symmetric variable orderings. The second question is how much larger an OBDD may be under the restriction to symmetric variable orderings. The performance ratio for some function f and symmetric variable orderings is defined by

$$R_f = \frac{\text{minimum OBDD size for } f \text{ and symmetric variable orderings}}{\text{minimum OBDD size for } f}.$$

For our example, as well as an example of Panda and Somenzi [13], the performance ratio converges to $4/3$. It is not known whether there is some nontrivial upper bound on the performance ratio.

Probably the most important problem is to characterize classes of functions for which all optimal variable orderings are nonsymmetric and to characterize subclasses where the performance ratio is close to 1. For such functions we may consider only symmetric variable orderings. The OBDD size for symmetric variable orderings is not much larger than the minimum OBDD size, and symmetric variable orderings can be found more easily since the search space is smaller. For example, the improved sifting algorithm can be used. We shall obtain a connection between the size of the symmetry sets and the property that all optimal variable orderings are nonsymmetric. In order to describe these results, we define $\text{PS}(a_1, \dots, a_d)$ as the set of all Boolean functions that are partially symmetric with respect to symmetry sets of the sizes $a_1, \dots, a_d$. Let $\text{PS}^d(a)$ denote the set of all Boolean functions that are partially symmetric with respect to d symmetry sets of size a each. We distinguish two basic cases.

The first case is that the symmetry sets have very different sizes. The simplest class where this occurs is $\text{PS}(1, n)$. For this class we prove two theorems.

Theorem 1. *For functions chosen randomly according to the uniform distribution from* $\text{PS}(1, n)$ *with probability* $1 - e^{-\Omega(n^{1/2})}$, *only nonsymmetric variable orderings are optimal.*

In fact, our example with a performance ratio converging to $4/3$ is a function in $\text{PS}(1, n)$. Nevertheless, the performance ratio of most functions in $\text{PS}(1, n)$ is close to 1.

Theorem 2. *For each* $\delta > 0$ *and for functions chosen randomly according to the uniform distribution from* $\text{PS}(1, n)$ *with probability* $1 - e^{-\Omega(n^{1/2})}$, *the performance ratio is smaller than* $1 + \delta$.

This theorem justifies the restriction to symmetric variable orderings for functions in $\text{PS}(1, n)$. We only remark that Theorems 1 and 2 also hold for all classes of partially symmetric functions with k symmetry sets of positive constant size and l symmetry sets of size $\Omega(n)$, where k and l are positive constants.

The second basic case is functions with symmetry sets of equal size.

Theorem 3. *Let d be a constant. For functions chosen randomly according to the uniform distribution from* $\text{PS}^d(n)$ *with probability* $1 - o(1)$, *only symmetric variable orderings are optimal.*

This is much stronger than the statement that with probability $1-o(1)$ there is some symmetric variable ordering that is optimal. Theorem 3 implies that with probability $1-o(1)$, the performance ratio is 1. Again the restriction to symmetric variable orderings is justified.

Theorem 3 complements the result of Sieling and Wegener [18]. They considered the classes $\mathrm{PS}^d(n)$ for constant n. These classes are motivated by the representation of functions in $\mathrm{PS}^d(n)$ by ordered multiple-valued decision diagrams (OMDDs). In [18] it is shown that for functions chosen randomly from $\mathrm{PS}^d(n)$ for constant n with probability $1-e^{-\Omega(d)}$, only symmetric variable orderings are optimal.

The paper is organized as follows. First we repeat the definition and some properties of OBDDs. A function with a performance ratio converging to $4/3$ is presented in Section 3. A method for the computation of the OBDD size for partially symmetric functions is described in Section 4. In Section 5 we show how to estimate the number of different contiguous submatrices of a random Boolean matrix. Then we use these results to prove the theorems.

2. DEFINITIONS AND PROPERTIES OF OBDDs

In this section we briefly repeat the definition and some properties of OBDDs. For a more detailed presentation and for applications of OBDDs, we refer to the survey articles of Wegener [20] and Bryant [3].

An OBDD is a directed acyclic graph with one source node. There are at most two sink nodes labeled by 0 and 1. Nonsink nodes, also called interior nodes, are labeled by a Boolean variable and have two outgoing edges, one labeled by 0 and the other one labeled by 1. Each input $a=(a_1,\dots,a_n)$ defines a computation path starting at the source. At an interior node labeled by x_i the path follows the a_i edge leaving this node. The label of the sink that is finally reached is equal to the value that the function represented by the OBDD takes for the input a.

The ordering condition for OBDDs requires the variables on each computation path to be tested according to a fixed ordering. In particular, each variable is tested at most once on each computation path. The size of an OBDD is the number of its interior nodes. The ith level of an OBDD denotes the set of all nodes labeled by the ith variable in the variable ordering.

Examples of OBDDs for the function $f(x_1,x_2,x_3)=x_1x_3\vee x_2$ are shown in Figure 1. The variable ordering is x_1,x_2,x_3. We usually omit the edge labels and use the convention that the edges leaving nodes on the left-hand side are labeled by 0 and the edges on the right-hand side are labeled by 1.

There are two reduction rules for OBDDs that reduce the size but do not change the represented function. By the deletion rule, each node v for which both successors are equal can be deleted, i.e., v is removed and all edges leading to v are redirected to the successor of v. By the merging rule, nodes v and w with the same label, the same 0-successor, and the same 1-successor are merged, i.e., w is removed and all edges leading to w are redirected to v. On a reduced OBDD (ROBDD) neither of the reduction rules can be applied. It is sometimes convenient to consider quasireduced OBDDs (QOBDDs) instead of ROBDDs. On QOBDDs the merging rule cannot be applied, and on each path from the source to

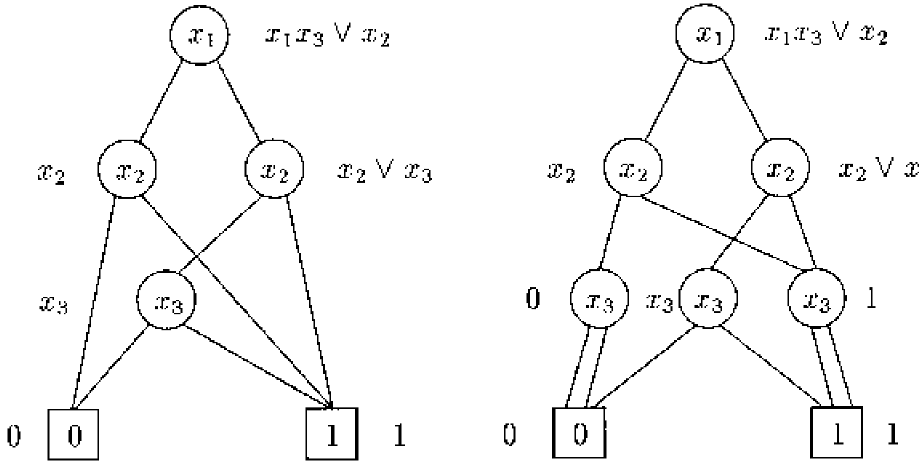

Fig. 1. An ROBDD and a QOBDD for the function $f(x_1, x_2, x_3) = x_1x_3 \vee x_2$.

the sink each variable is tested exactly once. A reduced OBDD can be converted into a quasireduced OBDD by applying the inverse deletion rule (see Fig. 1). The size increases by a factor of at most about n. Bryant [2] proved that ROBDDs for each function and each variable ordering are unique up to isomorphism. The same holds for QOBDDs. We remark that Theorems 1–3 also hold for QOBDDs.

Each node v of an OBDD computes a function f_v, namely, the function that is obtained when considering v as the source node of an OBDD. In Figure 1 these functions are given for each node. The following lemmas describe which functions are computed at the x_i nodes of an ROBDD and of a QOBDD. The lemmas are stated for the variable ordering $x_1, \ldots, x_n$. They can be applied to all variable orderings by renaming the variables. The first lemma was proved by Sieling and Wegener [17]. It is easy to adapt their proof for the quasireduced case.

Lemma 4. *Let T_i^* denote the set of different subfunctions $f_{|x_1=c_1,\ldots,x_{i-1}=c_{i-1}}$ of f that essentially depend on x_i (i.e., $f_{|x_1=c_1,\ldots,x_{i-1}=c_{i-1},\, x_i=0} \neq f_{|x_1=c_1,\ldots,x_{i-1}=c_{i-1},\, x_i=1}$). In the ROBDD for f and the variable ordering $x_1,\ldots,x_n$ there are exactly $|T_i^*|$ nodes labeled by x_i. For each function in T_i^* there is exactly one x_i node computing this function.*

Lemma 5. *Let S_i^* denote the set of different subfunctions $f_{|x_1=c_1,\ldots,x_{i-1}=c_{i-1}}$ of f. In the QOBDD for f and the variable ordering $x_1,\ldots,x_n$ there are exactly $|S_i^*|$ nodes labeled by x_i. For each function in S_i^* there is exactly one x_i node computing this function.*

3. A FUNCTION WITH A LARGE PERFORMANCE RATIO

The following example shows that by using nonsymmetric variable orderings, mergings of isomorphic parts of OBDDs may be possible that are not possible with symmetric variable orderings. This is the reason why nonsymmetric variable orderings may lead to smaller OBDDs. Let $n = 3k + 1$, $k \in \mathbb{N}$. The function f_n: $\{0,1\}^{n+1} \to \{0,1\}$ is defined by

$$f_n(y, x_1, \ldots, x_n) := \begin{cases} 1, & \text{if } \|x\|_1 = k, \\ y, & \text{if } \|x\|_1 = 2k+1, \\ 0, & \text{otherwise,} \end{cases}$$

where $\|x\|_1$ denotes the number of x variables equal to 1. Obviously, the symmetry sets are $\{y\}$ and $\{x_1, \ldots, x_n\}$. Figure 2 shows the ROBDDs G_1, G_2, and G_{opt} for f_{10} and the symmetric variable orderings $x_1, \ldots, x_{10}, y$ and $y, x_1, \ldots, x_{10}$ and the nonsymmetric variable ordering $x_1, \ldots, x_6, y, x_7, x_8, x_9, x_{10}$. In Figure 2 the tests of x variables are denoted by open circles and the tests of y by solid circles. Recall that the relative ordering of the x variables is irrelevant. Edges leading nowhere are abbreviations for edges leading to the 0-sink.

The trapezoid parts in the bottom of G_1 are isomorphic, but they cannot be merged because a test of y has to be performed only below the right trapezoid. The triangular parts in the top of G_2 cannot be merged because y is tested before. If we merge these parts, we forget the value of y. Therefore, it is reasonable to test y below the triangular part in the top and above the trapezoid part in the bottom. If we use such a nonsymmetric variable ordering, we need only one triangular part and one trapezoid part.

By the methods presented in Section 4 we can compute exact formulas for the number of nodes of OBDDs for f_n and both symmetric variable orderings, and also for the nonsymmetric variable ordering where y is tested after half of the x variables. These computations show that this nonsymmetric variable ordering is

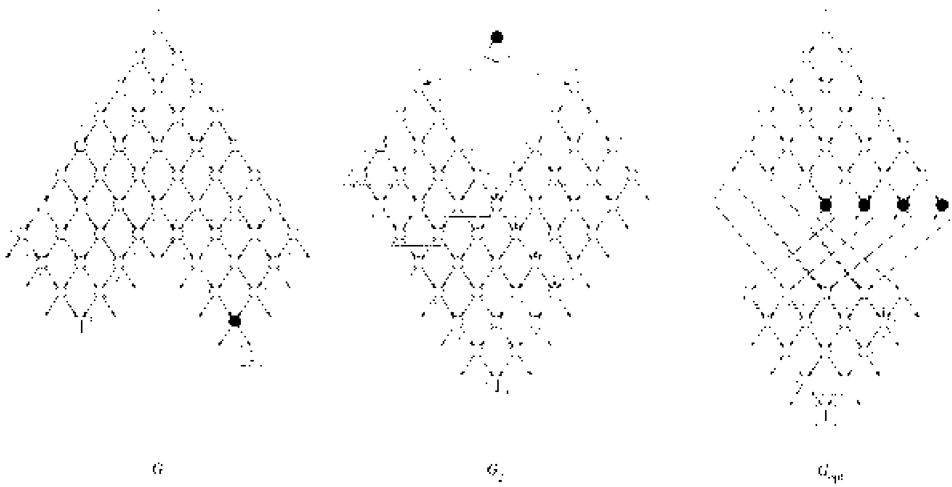

Fig. 2. ROBDDs G_1, G_2, and G_{opt} for the function f_{10}. The variable ordering of G_1 is $x_1, \ldots, x_{10}, y$, the variable ordering of G_2 is $y, x_1, \ldots, x_{10}$, and the variable ordering of G_{opt} is $x_1, \ldots, x_6, y, x_7, x_8, x_9, x_{10}$.

TABLE 1 Formulas for the ROBDD Size and QOBDD Size for f_n and Several Variable Orderings

Variable ordering	ROBDD size for f_n	QOBDD size for f_n
$x_1, \dots, x_n, y$	$\frac{1}{3}n^2 + \frac{4}{3}n + \frac{1}{3}$	$\frac{1}{3}n^2 + \frac{5}{3}n + 1$
$y, x_1, \dots, x_n$	$\frac{1}{3}n^2 + \frac{4}{3}n + \frac{4}{3}$	$\frac{1}{3}n^2 + 2n + \frac{2}{3}$
$x_1, \dots, x_{n/2}, y, x_{n/2+1}, \dots, x_n$ (n even)	$\frac{1}{4}n^2 + \frac{4}{3}n + \frac{2}{3}$	$\frac{1}{4}n^2 + 2n + 1$

optimal and that the performance ratio for this function converges to $4/3$. The formulas are given in Table 1.

4. THE STRUCTURE OF OBDDs FOR PARTIALLY SYMMETRIC FUNCTIONS

We are going to show how Lemmas 4 and 5 can be applied to partially symmetric functions. Hence, we show how to count the number of subfunctions described in Lemmas 4 and 5.

Partially symmetric functions have the property that the value of the function only depends on the number of 1s assigned to the variables in each symmetry set, but not on the position of these 1s. Therefore, we can represent a function f that is partially symmetric with respect to the symmetry sets $V_1, \dots, V_d$ by a d-dimensional matrix W_f, which we call the value matrix of f. Let $v_i = |V_i|$. Then W_f is a Boolean $(v_1 + 1) \times \cdots \times (v_d + 1)$ matrix. We index the rows in the ith direction from 0 to v_i. Then $W_f(i_1, \dots, i_d)$ is the value that f takes on all inputs with i_l variables of V_l equal to 1 (for $l = 1, \dots, d$). For totally symmetric functions, the value matrix is one-dimensional, i.e., the value vector considered by Wegener [19]. The value matrix of the function f_{10} in Section 3 is

		Number of x variables equal to 1										
		0	1	2	3	4	5	6	7	8	9	10
Number of y variables equal to 1	0	0	0	0	1	0	0	0	0	0	0	0
	1	0	0	0	1	0	0	0	1	0	0	0

It is easy to obtain the value matrix of some subfunction $f_{|x=c}$ from the value matrix of f. Let $x \in V_j$. If $c = 0$, we remove all entries $W_f(i_1, \dots, i_d)$ from W_f, where $i_j = v_j$. If $c = 1$, we remove all entries where $i_j = 0$. In this way we get a $(v_1 + 1) \times \cdots \times (v_{j-1} + 1) \times v_j \times (v_{j+1} + 1) \times \cdots \times (v_d + 1)$ matrix that is the value matrix of $f_{|x=c}$.

If we replace i_l variables from V_l ($l = 1, \dots, d$) by constants, the value matrix becomes a submatrix of W_f of size $(v_1 + 1 - i_1) \times \cdots \times (v_d + 1 - i_d)$. This submatrix of W_f consists of contiguous entries of W_f. We call such submatrices blocks of W_f. Each subfunction of f corresponds to a block of W_f and vice versa. Hence, we can count the number of different subfunctions of f that are obtained by replacing

i_l variables from V_l ($l = 1, \dots, d$) by counting the number of different blocks of size $(v_1 + 1 - i_1) \times \cdots \times (v_d + 1 - i_d)$ of the value matrix W_f. Together with Lemma 5, we get the following theorem describing the number of x_i nodes of QOBDDs for partially symmetric functions.

Theorem 6. *Let f be a partially symmetric function with the symmetry sets $V_1, \dots, V_d$ of sizes $v_1, \dots, v_d$. Let $i_l = |V_l \cap \{x_1, \dots, x_{i-1}\}|$, i.e., the number of variables from V_l that are arranged before x_i. Let $S(j_1, \dots, j_d)$ denote the number of different $j_1 \times \cdots \times j_d$ blocks of W_f. The QOBDD for f and the variable ordering $x_1, \dots, x_n$ contains exactly $S(v_1 + 1 - i_1, \dots, v_d + 1 - i_d)$ nodes labeled by x_i.*

In order to compute the total size of a QOBDD for the variable ordering $x_1, \dots, x_n$, we have to sum up some of the values $S(i_1, \dots, i_d)$. We arrange the values $S(i_1, \dots, i_d)$ in a directed d-dimensional grid graph of size $(v_1 + 1) \times \cdots \times (v_d + 1)$. The node set of the grid graph is $V = \{(i_1, \dots, i_d) | i_j \in \{1, \dots, v_j + 1\}, j = 1, \dots, d\}$. For each node $(i_1, \dots, i_d)$ and each $j \in \{1, \dots, d\}$ with $i_j \geq 2$ there is an edge leading in the jth direction to the node $(i_1, \dots, i_{j-1}, i_j - 1, i_{j+1}, \dots, i_d)$. Each node $(i_1, \dots, i_d) \in V$ is labeled by $S(i_1, \dots, i_d)$. The source of the grid graph is the node $(v_1 + 1, \dots, v_d + 1)$. It is labeled by $S(v_1 + 1, \dots, v_d + 1)$, i.e., the number of nodes of the first level of the QOBDD. If at this level the variable $x_1 \in V_j$ is tested, we follow the edge leading in the jth direction to the node $(v_1 + 1, \dots, v_{j-1} + 1, v_j, v_{j+1} + 1, \dots, v_d + 1)$. The label of this node is equal to the number of nodes of the second level of the QOBDD. In this way we can follow a path through the grid graph corresponding to the variable ordering of the QOBDD. This path ends at the sink $(1, \dots, 1)$ of the grid graph. $S(1, \dots, 1)$ is the number of different values that the function takes, i.e., the number of sinks. By summing up the labels along this path [except $S(1, \dots, 1)$], we get the number of interior nodes of the QOBDD.

The grid graph for the example f_{10} from Section 3 and the size of its QOBDDs is shown in Figure 3a.

We see that after computing the values $S(i_1, \dots, i_d)$, we can compute the size of a QOBDD for each variable ordering in time $O(n)$, where n is the number of variables. On the other hand, we can compute a variable ordering leading to

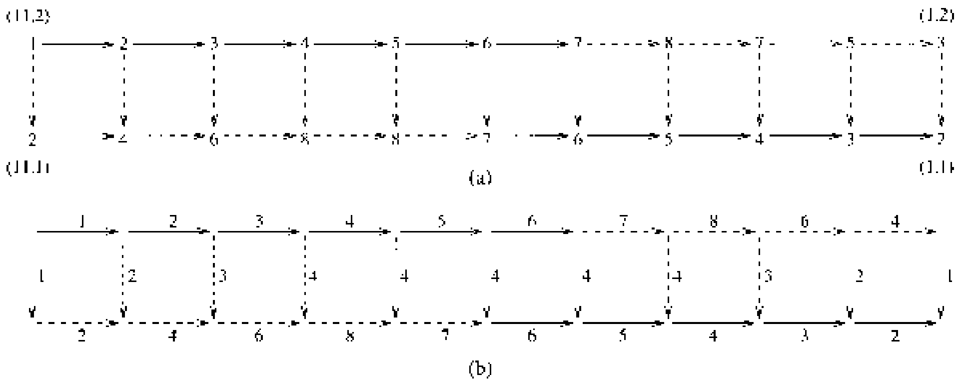

Fig. 3. Grid graphs for the function f_{10} and the size of (a) QOBDDs and (b) ROBDDs.

minimum QOBDD size. We have just to determine a shortest path from the source of the grid graph to its sink. This is possible in linear time with respect to the size of the grid graph by a simple dynamic programming approach. In Figure 3, edges belonging to some shortest path are drawn as solid lines, while all other edges are drawn as dashed lines. We see that exactly those variable orderings are optimal where y is tested after five or six of the x variables. The QOBDD size is 46.

If we consider functions with n symmetry sets of size 1 each, the grid graph is an n-dimensional hypercube, i.e., it has size 2^n. Again it is possible in time $O(n)$ to compute the OBDD size for each variable ordering after computing the labels of the grid graph. However, in this case, even the computation of the labels of the grid graph seems to be hard.

Now we generalize the results to ROBDDs. By Lemma 4 there are x_i nodes only for those subfunctions $f_{|x_1=c_1,\dots,x_{i-1}=c_{i-1}}$ that essentially depend on x_i. Therefore, the number of nodes of the ith level of ROBDDs is also determined by the choice of the variable tested at the ith level rather than merely by the choice of the variables tested at the levels $1,\dots,i-1$ as in the case of QOBDDs. For this reason, we get grid graphs with edge labels instead of node labels.

We describe how to compute the label of the edge from the node $(i_1,\dots,i_d)$ to the node $(i_1,\dots,i_{j-1},i_j-1,i_{j+1},\dots,i_d)$. Here we have the situation that some variable in V_j is tested. Therefore, we have to count only those $i_1\times\cdots\times i_d$ blocks W_g of W_f that represent subfunctions g that essentially depend on the variables in V_j. (It is obvious that g either essentially depends on all variables in V_j or does not essentially depend on any variable in V_j.) We use the notion of rows of W_g in the jth direction. Such a row consists of the entries $W_g[p_1,\dots,p_{j-1},0,p_{j+1},\dots,p_d]$, $W_g[p_1,\dots,p_{j-1},1,p_{j+1},\dots,p_d],\dots$. If the entries of each such row are equal, a change of variables in V_j cannot change the value that g takes. On the other hand, if there is some row where two entries are not equal, a change of some variable in V_j may change the value of g. Therefore, we have to count only those $i_1\times\cdots\times i_d$ blocks of W_f that contain some nonconstant row in the jth direction.

Theorem 7. *Let f be a partially symmetric function with the symmetry sets $V_1,\dots,V_d$ of sizes $v_1,\dots,v_d$. Let $i_l=|V_l\cap\{x_1,\dots,x_{i-1}\}|$, i.e., the number of variables from V_l that are arranged before x_i. Let $T_j(j_1,\dots,j_d)$ denote the number of different $j_1\times\cdots\times j_d$ blocks of W_f that have some nonconstant row in the jth direction. Let $x_i\in V_{j^*}$. The ROBDD for f and the variable ordering $x_1,\dots,x_n$ contains exactly $T_{j^*}(v_1+1-i_1,\dots,v_d+1-i_d)$ nodes labeled by x_i.*

The grid graph for ROBDDs and the function f_{10} from Section 3 is shown in Figure 3b. The computation of the ROBDD size for some variable ordering is done by adding the edge labels along the same path as in the case of QOBDDs. Again the minimization can be done by computing a shortest path from the source to the sink.

Finally, we remark that we may get larger symmetry sets and, therefore, a more efficient minimization algorithm by considering negative symmetries. Now for each variable x_i, the complement $\bar{x}_i$ of x_i may be contained in some set V_j instead of x_i. This has the following meaning: If $V_j=\{x_{i_1},\dots,x_{i_l},\bar{x}_{j_1},\dots,\bar{x}_{j_m}\}$, the value of the function f only depends on the number of 1s among $x_{i_1},\dots,x_{i_l},\bar{x}_{j_1},\dots,\bar{x}_{j_m}$, but it does not depend on the position of these 1s. By substituting $\bar{x}_{j_1},\dots,\bar{x}_{j_m}$ by

$x_{j_1}, \ldots, x_{j_m}$ we get a new function f^* with only (positive) symmetries. Now we may apply the results developed above. From an OBDD for f^* we get an OBDD for f by exchanging the 0- and the 1-successor of all nodes labeled by $x_{j_1}, \ldots, x_{j_m}$.

5. ESTIMATES FOR THE LABELS OF THE GRID GRAPH OF RANDOM PARTIALLY SYMMETRIC FUNCTIONS

In this section we show how to estimate the labels of a grid graph of a random partially symmetric function. First we mention how to choose a function from some class $\mathrm{PS}(a_1, \ldots, a_d)$ according to the uniform distribution. Recall that the value matrix of such a function is a Boolean $(a_1 + 1) \times \cdots \times (a_d + 1)$ matrix. We choose each entry of an $(a_1 + 1) \times \cdots \times (a_d + 1)$ matrix independently and with probability $1/2$ for 0 and probability $1/2$ for 1. This leads to a uniform distribution since for each function in $\mathrm{PS}(a_1, \ldots, a_d)$ there is exactly one value matrix and vice versa.

The grid graph corresponding to such a random function can be partitioned into three parts. The estimates for the labels in each part are obtained by different methods.

Top Part. The labels in the top part describe the number of different "large" blocks of the value matrix. It is intuitively clear that with high probability, all such blocks are different and that no such block only consists of constant rows in some direction. Hence, with high probability for all nodes $(i_1, \ldots, i_d)$ in the top part of the grid graph and all j, where $i_j \geq 2$, we have $S(i_1, \ldots, i_d) = T_j(i_1, \ldots, i_d) = (a_1 + 1 - i_1) \cdot \cdots \cdot (a_d + 1 - i_d) =: h(i_1, \ldots, i_d)$, i.e., the number of $i_1 \times \cdots \times i_d$ blocks of an $(a_1 + 1) \times \cdots \times (a_d + 1)$ matrix.

Bottom Part. The labels in the bottom part describe the number of different "small" blocks of the value matrix. It is intuitively clear that with high probability, all possible assignments of small blocks do occur in the value matrix. Hence, with high probability for all nodes $(i_1, \ldots, i_d)$ in the bottom part we have $S(i_1, \ldots, i_d) = 2^{i_1 \cdot \cdots \cdot i_d} =: g(i_1, \ldots, i_d)$, i.e., the number of possible assignments of an $i_1 \times \cdots \times i_d$ block. Furthermore, for all $j \in \{1, \ldots, d\}$, where $i_j \geq 2$, with high probability $T_j(i_1, \ldots, i_d) = 2^{i_1 \cdot \cdots \cdot i_d} - 2^{i_1 \cdot \cdots \cdot i_{j-1} i_{j+1} \cdot \cdots \cdot i_d} =: g_j(i_1, \ldots, i_d)$, i.e., the number of possible assignments of an $i_1 \times \cdots \times i_d$ block that do not only consist of constant rows in the jth direction.

Middle Part. In the middle part of the grid graph the labels may be much smaller than their upper bounds $h(i_1, \ldots, i_d)$, $g(i_1, \ldots, i_d)$, and $g_j(i_1, \ldots, i_d)$, respectively. Furthermore, the variance is large, but for the case $\mathrm{PS}(1, n)$ we do not need estimates for the labels in the middle part and for the case $\mathrm{PS}^d(n)$ weaker estimates are sufficient.

We conclude that OBDDs for random partially symmetric functions have a top part and a bottom part that with high probability are of their maximum possible size. However, the middle part may be smaller than its maximum possible size. Wegener [21] obtained a similar partition for OBDDs for arbitrary random functions.

In order to simplify the notation, we consider in the following the class $\mathrm{PS}^d(n-1)$ instead of $\mathrm{PS}^d(n)$. Then the value matrix M is a d-dimensional $n\times\cdots\times n$ matrix. Furthermore, we have $h(i_1,\dots,i_d)=(n+1-i_1)\cdot\cdots\cdot(n+1-i_d)$. The results for $\mathrm{PS}(1,n)$ can be obtained in a similar way. Hence, we only state these results without proof.

We start with the top part of the grid graph. Let $\varepsilon\in(0,1)$. The top part consists of all nodes $(i_1,\dots,i_d)$, where $i_1\cdot\cdots\cdot i_d\geq(2d+\varepsilon)\log n$. Let (E1) be the event that for all nodes $(i_1,\dots,i_d)$ of the top part we have $S(i_1,\dots,i_d)=h(i_1,\dots,i_d)$. Let (E2) be the event that for all nodes $(i_1,\dots,i_d)$ of the top part and all $j\in\{1,\dots,d\}$, where $i_j\geq 2$, we have $T_j(i_1,\dots,i_d)=S(i_1,\dots,i_d)$.

Lemma 8. *Let* $\varepsilon\in(0,1)$. *Then* $\mathrm{Prob(E1)}\geq 1-[(2d+2)\log n]^d n^{-\varepsilon}$.

Proof. First we consider only a set I of nodes in the bottom of the top part and prove that with high probability for all $(i_1,\dots,i_d)\in I$ it holds that $S(i_1,\dots,i_d)=h(i_1,\dots,i_d)$. In a second step we show that this implies $S(i_1,\dots,i_d)=h(i_1,\dots,i_d)$ for all nodes $(i_1,\dots,i_d)$ of the top part. Let

$$I=\Big\{(i_1,\dots,i_d)\Big| i_1\cdot\cdots\cdot i_d\geq(2d+\varepsilon)\log n\wedge\forall j\in\{1,\dots,d\}: i_j\leq(2d+2)\log n\Big\}.$$

Obviously $|I|\leq[(2d+2)\log n]^d$. We estimate the probability that there is some node $(i_1,\dots,i_d)$ with $S(i_1,\dots,i_d)<h(i_1,\dots,i_d)$. There are $\binom{h(i_1,\dots,i_d)}{2}$ pairs consisting of $i_1\times\cdots\times i_d$ blocks of M. The probability that such a pair consists of equal blocks is $2^{-i_1\cdot\,\cdots\,\cdot i_d}=1/g(i_1,\dots,i_d)$. This is obvious if the blocks of a pair are disjoint. If the blocks A and B of the pair (A,B) overlap, then $A=B$ implies that the blocks are periodic. For the two-dimensional case, this situation is depicted in Figure 4. Let $j_1\times\cdots\times j_d$ be the size of the block shared by A and B. Then there are $2^{2i_1\cdot\,\cdots\,\cdot i_d-j_1\cdot\,\cdots\,\cdot j_d}$ possible assignments to A and B and $2^{i_1\cdot\,\cdots\,\cdot i_d-j_1\cdot\,\cdots\,\cdot j_d}$ possible assignments where A and B are equal. Again the probability of $A=B$ is $2^{-i_1\cdot\,\cdots\,\cdot i_d}$.

Altogether, the probability that for some fixed $(i_1,\dots,i_d)\in I$ there is some pair consisting of equal $i_1\times\cdots\times i_d$ blocks is $\binom{h(i_1,\dots,i_d)}{2}/g(i_1,\dots,i_d)\leq n^{-\varepsilon}$. Hence, with a probability of at least $1-|I|n^{-\varepsilon}$, there is no pair of equal $i_1\times\cdots\times i_d$ blocks for all $(i_1,\dots,i_d)\in I$. In other words, all such blocks are different. This implies the claim for all $(i_1,\dots,i_d)\in I$.

Now we drop the restriction $i_j\leq(2d+2)\log n$. Then for each node $(i'_1,\dots,i'_d)$ of the top part that is not contained in I, there is some node $(i_1,\dots,i_d)\in I$ that is componentwise not larger than $(i'_1,\dots,i'_d)$. Hence, it suffices to prove that $S(i_1,\dots,i_d)=h(i_1,\dots,i_d)$ implies $S(i'_1,\dots,i'_d)=h(i'_1,\dots,i'_d)$. Assume that $S(i'_1,\dots,i'_d)<h(i'_1,\dots,i'_d)$. This implies that some $i'_1\times\cdots\times i'_d$ block occurs more than once in M. Therefore, also some $i_1\times\cdots\times i_d$ block occurs more than once, in contradiction to $S(i_1,\dots,i_d)=h(i_1,\dots,i_d)$. ∎

Lemma 9. *Let* $\varepsilon\in(0,1)$. *Then* $\mathrm{Prob(E2)}\geq 1-d[(2d+2)\log n]^d n^{-\varepsilon/2}$.

Proof. Again we consider in the first step only the nodes contained in the set I. Let $(i_1,\dots,i_d)\in I$. Let $j\in\{1,\dots,d\}$ and $i_j\geq 2$. The number of assignments to an

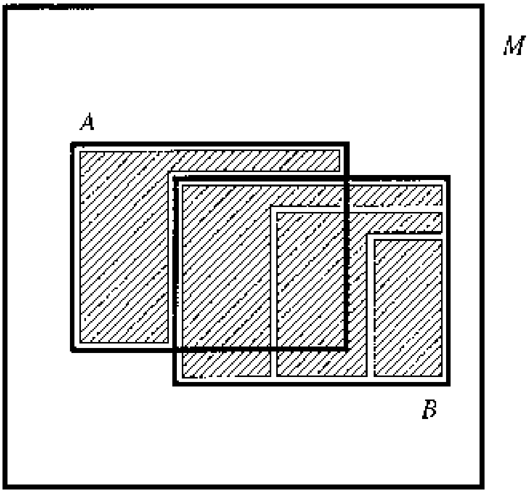

Fig. 4. The situation where blocks A and B overlap. If $A = B$, then all the hatched areas are equal to the left upper part of A.

$i_1 \times \cdots \times i_d$ block that only consist of constant rows in the jth direction is $2^{i_1 \cdot \cdots \cdot i_{j-1} i_{j+1} \cdot \cdots \cdot i_d}$. Hence, the probability that an $i_1 \times \cdots \times i_d$ block of M only consists of constant rows in the jth direction is $2^{i_1 \cdot \cdots \cdot i_{j-1} i_{j+1} \cdot \cdots \cdot i_d} / 2^{i_1 \cdot \cdots \cdot i_d} = 2^{i_1 \cdot \cdots \cdot i_d(-1+1/i_j)} \leq 2^{-i_1 \cdot \cdots \cdot i_d/2} = g(i_1, \ldots, i_d)^{-1/2}$. The probability that for a fixed $(i_1, \ldots, i_d) \in I$ there is a block that only consists of constant rows in some direction is bounded by $h(i_1, \ldots, i_d) g(i_1, \ldots, i_d)^{-1/2} d \leq d n^{-\varepsilon/2}$. The reason for the extra factor d is that there are at most d possible choices of j. Hence, the probability that there is some $i_1 \times \cdots \times i_d$ block, where $(i_1, \ldots, i_d) \in I$, that only consists of constant rows in some direction is bounded by $d|I| n^{-\varepsilon/2}$. The generalization to all $(i_1, \ldots, i_d)$ of the top part can be done in the same way as in the proof of Lemma 8. ■

Now we consider the bottom part. Estimates for the labels in the bottom part are only used for the case $\mathrm{PS}(1, n)$. For the sake of completeness, we also present the results for $\mathrm{PS}^d(n)$. The bottom part consists of all nodes $(i_1, \ldots, i_d)$ of the grid graph, where $i_1 \cdot \cdots \cdot i_d \leq (d - \varepsilon) \log n$. Let (E3) be the event that for all nodes $(i_1, \ldots, i_d)$ of the bottom part we have $S(i_1, \ldots, i_d) = g(i_1, \ldots, i_d)$. Let (E4) be the event that for all nodes $(i_1, \ldots, i_d)$ of the bottom part and all j, where $i_j \geq 2$, we have $T_j(i_1, \ldots, i_d) = g_j(i_1, \ldots, i_d)$.

Lemma 10. *Let* $\varepsilon \in (0, 1)$. *Then* $\mathrm{Prob(E3)} \geq 1 - e^{-\Omega(n^{\varepsilon}/\log n)}$ *and* $\mathrm{Prob(E4)} \geq 1 - e^{-\Omega(n^{\varepsilon}/\log n)}$.

Proof. In order to prove the estimate of Prob(E3), we consider the random choice of the value matrix as an experiment where balls are thrown randomly into buckets. Let $(i_1, \ldots, i_d)$ belong to the bottom part. We partition M into $b(i_1, \ldots, i_d) := \lfloor n/i_1 \rfloor \cdot \cdots \cdot \lfloor n/i_d \rfloor$ disjoint blocks of size $i_1 \times \cdots \times i_d$. For each of these disjoint blocks, there is a ball. Hence, there are $b(i_1, \ldots, i_d)$ balls. For each possible assignment to such a block, there is a bucket. Altogether, there are $g(i_1,\dots,i_d)$ buckets. The assignment of a block is chosen by throwing the corresponding ball randomly into a bucket. Since the blocks are disjoint, the throws for each block are independent. Let Y be the random variable describing the number of nonempty buckets after performing the experiment. Then Y is a lower bound for the number of different blocks. The mean value of Y is given by the formula (which is due to Kolchin, Sevast'yanov, and Christyakov [8])

$$E[Y] = g(i_1,\dots,i_d) - g(i_1,\dots,i_d)e^{-\alpha} + \frac{\alpha}{2}e^{-\alpha} - O\left(\frac{\alpha(1+\alpha)}{g(i_1,\dots,i_d)}e^{-\alpha}\right),$$

where $\alpha = b(i_1,\dots,i_d)/g(i_1,\dots,i_d)$. From $i_1 \cdot \cdots \cdot i_d \le (d-\varepsilon)\log n$, it follows $\alpha = \Omega(n^{\varepsilon}/\log n)$ and $E[Y] \ge g(i_1,\dots,i_d) - n^{d-\varepsilon}e^{-\Omega(n^{\varepsilon}/\log n)}$, i.e., $E[Y]$ converges exponentially to the upper bound $g(i_1,\dots,i_d)$ on Y. Since Y is always an integer, we conclude that $\mathrm{Prob}(Y < g(i_1,\dots,i_d)) = e^{-\Omega(n^{\varepsilon}/\log n)}$. Since there are less than $(d-\varepsilon)^d \log^d n$ nodes in the bottom part, the claim for (E3) follows.

Since with probability $1 - e^{-\Omega(n^{\varepsilon}/\log n)}$ all $i_1 \times \cdots \times i_d$ blocks occur in M, with the same probability all $i_1 \times \cdots \times i_d$ blocks occur that do not only consist of constant rows in some direction. This implies the claim for (E4). ■

Finally, we consider the middle part, which consists of all nodes not belonging to the top part nor to the bottom part. Let $(i_1,\dots,i_d)$ be a fixed node in the middle part. Similar to the proofs for the nodes in the top part, we consider pairs of $i_1 \times \cdots \times i_d$ blocks. Let $X(i_1,\dots,i_d)$ be the random variable describing the number of pairs consisting of equal blocks. We are going to derive estimates for $X(i_1,\dots,i_d)$. Such estimates imply estimates for $S(i_1,\dots,i_d)$ by the following lemma.

Lemma 11. *For all* $(i_1,\dots,i_d) \in \{1,\dots,n\}^d$: $S(i_1,\dots,i_d) \ge h(i_1,\dots,i_d) - X(i_1,\dots,i_d)$.

Proof. If $X(i_1,\dots,i_d) = 0$, then obviously $S(i_1,\dots,i_d) = h(i_1,\dots,i_d)$. This means that there are $h(i_1,\dots,i_d)$ equivalence classes of blocks where each equivalence class consists of a single block. For each pair consisting of equal blocks, at most two of these equivalence classes are merged. This implies the claim. ■

In order to estimate $X(i_1,\dots,i_d)$, we apply Chebyshev's inequality. Hence, we first derive formulas for the mean value and the variance of $X(i_1,\dots,i_d)$. Let p be the number of pairs of $i_1 \times \cdots \times i_d$ blocks of M. Because of $h(i_1,\dots,i_d) \le n^d$, we have $p \le n^{2d}/2$. Now we define random variables $X_1,\dots,X_p$. For each pair of $i_1 \times \cdots \times i_d$ blocks there is such a random variable, which takes the value 1 if the blocks of the corresponding pair are equal and otherwise the value 0. By the proof of Lemma 8 we have for all $i \in \{1,\dots,p\}$: $E[X_i] = 2^{-i_1 \cdot \cdots \cdot i_d}$. Because of $X(i_1,\dots,i_d) = X_1 + \cdots + X_p$, we have

$$E[X(i_1,\dots,i_d)] = p2^{-i_1 \cdot \cdots \cdot i_d} \le \frac{n^{2d}}{2}2^{-i_1 \cdot \cdots \cdot i_d}.$$

In order to compute the variance of $X(i_1,\dots,i_d)$, we consider $E[X(i_1,\dots,i_d)^2] = \sum_{1 \le i,j \le p} E[X_i X_j]$. Let (A,B) be the pair of blocks belonging to X_i and let (C,D) be the pair of blocks belonging to X_j.

Case 1. C does not overlap A or B, nor does D overlap A or B. Then X_i and X_j are independent and $E[X_iX_j] = E[X_i]E[X_j] = 2^{-2i_1 \cdot \dots \cdot i_d}$. We estimate the number of such terms of the sum by p^2.

Case 2. C overlaps A or B, and D overlaps A or B. Then $E[X_iX_j] \le E[X_i] = 2^{-i_1 \cdot \dots \cdot i_d}$. We estimate the number of terms of the sum where Case 2 occurs. The number of possible positions of A and B in M is p. The number of possible positions of C, where C overlaps A, is bounded by $(2i_1) \cdot \dots \cdot (2i_d)$. Since C may overlap A or B, the number of positions of C is bounded by $2 \cdot (2i_1) \cdot \dots \cdot (2i_d)$. The same holds for the number of possible positions of D. Altogether, the number of terms of the sum for which Case 2 occurs is at most $p \cdot 2^{2d+2} \cdot i_1^2 \cdot \dots \cdot i_d^2$.

We conclude $E[X(i_1, \dots, i_d)^2] \le p^2 2^{-2i_1 \cdot \dots \cdot i_d} + p \cdot 2^{2d+2} \cdot i_1^2 \cdot \dots \cdot i_d^2 \cdot 2^{-i_1 \cdot \dots \cdot i_d}$. Then

$$
\begin{aligned}
V[X(i_1,\dots,i_d)] &= E\left[X(i_1,\dots,i_d)^2\right] - E[X(i_1,\dots,i_d)]^2 \\
&\le 2^{2d+1} \cdot n^{2d} \cdot i_1^2 \cdot \dots \cdot i_d^2 \cdot 2^{-i_1 \cdot \dots \cdot i_d}.
\end{aligned}
$$

We do not estimate $X(i_1, \dots, i_d)$ for all $(i_1, \dots, i_d)$ of the middle part. For our purposes it suffices to consider only those $(i_1, \dots, i_d)$ of the middle part that fulfill the following additional conditions. These extra conditions are derived in Section 7.

(P1) It holds that $i_1 + \dots + i_d = d - 1 + s$, where $(2d + \varepsilon)\log n > s > (d - \varepsilon)\log n$.

(P2) There are at least two components of $(i_1, \dots, i_d)$ that are different from 1.

Let (E5) be the event that for all nodes $(i_1, \dots, i_d)$ of the middle part fulfilling (P1) and (P2), it holds that $S(i_1, \dots, i_d) \ge h(i_1, \dots, i_d) - 4n$.

Lemma 12. *Let* $\varepsilon \in (0, 2/5)$. *Then* $\operatorname{Prob}(\text{E5}) = 1 - o(1)$.

Proof. We are going to prove the bound on the probability of (E5) by bounding the probability of $X(i_1, \dots, i_d) \ge 4n$ using Chebyshev's inequality and by applying Lemma 11. Let $\gamma = n^{2d+1-2\varepsilon} 2^{-i_1 \cdot \dots \cdot i_d - 1}$. First we estimate $E[X(i_1, \dots, i_d)] + \gamma$ by $4n$. It holds $E[X(i_1, \dots, i_d)] + \gamma \le n^{2d+1-2\varepsilon} 2^{-i_1 \cdot \dots \cdot i_d}$. In order to get an upper bound for this term, we maximize this term under the conditions (P1) and (P2) and the condition that $(i_1, \dots, i_d)$ belongs to the middle part. Equivalently, we may minimize $i_1 \cdot \dots \cdot i_d$ under these conditions. The product $i_1 \cdot \dots \cdot i_d$ takes its minimum value under these conditions iff $d - 2$ factors are equal to 1, one factor is equal to 2, and one factor is equal to $s - 1$. We conclude $E[X(i_1, \dots, i_d)] + \gamma \le n^{2d+1-2\varepsilon} 2^{(-2)(s-1)} \le n^{2d+1-2\varepsilon} 2^{(-2)(d-\varepsilon)\log n + 2} = 4n$. The second inequality follows from $s > (d - \varepsilon)\log n$. Therefore, by Chebyshev's inequality we get $\operatorname{Prob}(X(i_1, \dots, i_d) \ge 4n) \le \operatorname{Prob}(X(i_1, \dots, i_d) \ge E[X(i_1, \dots, i_d)] + \gamma) \le V[X(i_1, \dots, i_d)]/\gamma^2 = O(\log^2 n / n^{2-5\varepsilon})$. Since the number of nodes $(i_1, \dots, i_d)$ in the middle part is less than $[(2d + \varepsilon)\log n]^d$, with probability $1 - O(\log^{d+2} n / n^{2-5\varepsilon})$ we have $X(i_1, \dots, i_d) \le 4n$. Together with Lemma 11, the claim follows. ■

Now we estimate the number of $i_1 \times \cdots \times i_d$ blocks that only consist of constant rows in some direction and where $(i_1, \ldots, i_d)$ is in the middle part. Let (E6) be the event that for all $(i_1, \ldots, i_d)$ of the middle part and an $j \in \{1, \ldots, d\}$, where $i_j \geq 2$, we have $T_j(i_1, \ldots, i_d) \geq S(i_1, \ldots, i_d) - 2n^{d/2+\varepsilon/2}$.

Lemma 13. *Let* $\varepsilon \in (0, 1)$. *Then* $\text{Prob(E6)} = 1 - o(1)$.

Proof. Again we apply Chebyshev's inequality in order to estimate the probability that there are many blocks that are constant in some direction. Let $(i_1, \ldots, i_d)$ belong to the middle part. Let some direction $j \in \{1, \ldots, d\}$, where $i_j \geq 2$, be fixed and let $a := h(i_1, \ldots, i_d)$. We define random variables $Z_1, \ldots, Z_a$. For each $i_1 \times \cdots \times i_d$ block there is one of these random variables. It takes the value 1 if the block only consists of constant rows in the jth direction, and 0 otherwise. Let $Z(i_1, \ldots, i_d) = Z_1 + \cdots + Z_a$, i.e., the random variable describing the number of $i_1 \times \cdots \times i_d$ blocks that only consist of constant rows in the jth direction. We have $E[Z_i] = \text{Prob}(Z_i = 1) = 2^{i_1 \cdot \cdots \cdot i_{j-1} i_{j+1} \cdot \cdots \cdot i_d} / 2^{i_1 \cdot \cdots \cdot i_d} = 2^{(-1+1/i_j) i_1 \cdot \cdots \cdot i_d} \leq n^{-d/2+\varepsilon/2}$ and, hence,

$$E[Z(i_1, \ldots, i_d)] = aE[Z_1] \leq an^{-d/2+\varepsilon/2} \leq n^{d/2+\varepsilon/2}.$$

The last inequality follows from $a \leq n^d$. Now we estimate the variance of $Z(i_1, \ldots, i_d)$ by considering $E[Z(i_1, \ldots, i_d)^2] = \sum_{1 \leq i, j \leq a} E[Z_i Z_j]$.

Case 1: The blocks corresponding to Z_i and Z_j are disjoint. Then $E[Z_i Z_j] = E[Z_1]^2$. The number of such terms of the sum is bounded by a^2.

Case 2: The blocks corresponding to Z_i and Z_j overlap. Then $E[Z_i Z_j] \leq E[Z_1] \leq n^{-d/2+\varepsilon/2}$. The number of possible positions of the block corresponding to Z_i is a; the number of possible positions of the block corresponding to Z_j is bounded by $(2i_1) \cdot \cdots \cdot (2i_d)$, since the blocks overlap.

Hence, $E[Z(i_1, \ldots, i_d)^2] \leq a^2 E[Z_1]^2 + a \cdot 2^d \cdot i_1 \cdot \cdots \cdot i_d \cdot n^{-d/2+\varepsilon/2}$ and, therefore, $V[Z(i_1, \ldots, i_d)] = E[Z(i_1, \ldots, i_d)^2] - E[Z(i_1, \ldots, i_d)]^2 \leq a \cdot 2^d \cdot i_1 \cdot \cdots \cdot i_d \cdot n^{-d/2+\varepsilon/2}$. By Chebyshev's inequality we get

$$\text{Prob}\big(Z(i_1, \ldots, i_d) \geq E[Z(i_1, \ldots, i_d)] + n^{d/2+\varepsilon/2}\big)$$
$$\leq \frac{V[Z(i_1, \ldots, i_d)]}{n^{d+\varepsilon}} \leq \frac{2^d \cdot i_1 \cdot \cdots \cdot i_d}{n^{d/2+\varepsilon/2}} = O\left(\frac{\log n}{n^{d/2+\varepsilon/2}}\right).$$

Again we used the fact that $a \leq n^d$. Furthermore, $E[Z(i_1, \ldots, i_d)] + n^{d/2+\varepsilon/2} \leq 2n^{d/2+\varepsilon/2}$. The claim follows since there are at most $(2d+\varepsilon)^d \log^d n$ nodes in the middle part and at most d possible directions. ■

We conclude this section with the results for PS$(1, n)$. Now the value matrix is of size $2 \times (n+1)$. The number of $i_1 \times i_2$ blocks in a $2 \times (n+1)$ matrix is $h(i_1, i_2) = (3 - i_1)(n + 2 - i_2)$.

Let (A1) be the event that for all $(i_1, i_2) \in \{1, 2\} \times \{1, \ldots, n+1\}$, where $i_1 \cdot i_2 \geq n^{1/2}$, it holds that $S(i_1, i_2) = h(i_1, i_2)$. Let (A2) be the event that for all $(i_1, i_2) \in \{1, 2\} \times \{1, \ldots, n+1\}$, where $i_1 \cdot i_2 \geq n^{1/2}$, and for all $j \in \{1, 2\}$, where $i_j \geq 2$, it holds that $T_j(i_1, i_2) = S(i_1, i_2)$.

Lemma 14. $\text{Prob(A1)} = 1 - e^{-\Omega(n^{1/2})}$ *and* $\text{Prob(A2)} = 1 - e^{-\Omega(n^{1/2})}$.

The probabilities of (A1) and (A2) are even exponentially close to 1. The reason for this is that the top part according to the definition of (A1) and (A2) is smaller since the bound on $i_1 \cdot i_2$ is $n^{1/2}$ rather than $c \cdot \log n$. Then the probability that some pair consists of equal blocks is even exponentially small.

For the bottom part we define (A3) as the event that for all $(i_1, i_2) \in \{1,2\} \times \{1, \ldots, n+1\}$, where $i_1 \cdot i_2 \leq (1-\varepsilon)\log n$, it holds that $S(i_1, i_2) = g(i_1, i_2) = 2^{i_1 i_2}$. Let (A4) be the event that for all $(i_1, i_2) \in \{1,2\} \times \{1, \ldots, n+1\}$, where $i_1 \cdot i_2 \leq (1-\varepsilon)\log n$, and all $j \in \{1,2\}$, where $i_j \geq 2$, it holds that $T_j(i_1, i_2) = g_j(i_1, i_2) = 2^{i_1 i_2} - 2^{i_1 i_2 / i_j}$.

Lemma 15. *Let* $\varepsilon \in (0,1)$. *Then* $\text{Prob(A3)} = 1 - e^{-\Omega(n^{\varepsilon}/\log n)}$ *and* $\text{Prob(A4)} = 1 - e^{-\Omega(n^{\varepsilon}/\log n)}$.

6. PROOFS OF THEOREMS 1 AND 2

We consider a random partially symmetric function chosen from $\text{PS}(1, n)$ according to the uniform distribution. We choose $\varepsilon > 1/2$ and assume that the events (A1)–(A4) occur for this function. This is true with probability $1 - e^{-\Omega(n^{1/2})}$. We only present the proof for ROBDDs, since the proof for QOBDDs works in the same way. The grid graph for the function is shown in Figure 5. The edge labels shown in Figure 5 follow from (A1)–(A4). For the proof of Theorem 1, we only consider the first three levels of the grid graph, which are shown in Figure 5a, and the last three levels shown in Figure 5d. The paths corresponding to the symmetric variable orderings are π_1 and π_4. It is easy to see that π_1^* and π_4^* are shorter than π_1 and π_4, respectively. Hence, both symmetric variable orderings are not optimal. This implies Theorem 1.

For the proof of Theorem 2, we discuss where in an optimal variable ordering the single variable of the first symmetry set has to be tested. This is equivalent to the question where does the vertical edge occur in a shortest path of the grid graph. The following inequalities are easy to verify:

$$\forall i \in \{2, \ldots, n\}: h(2, i-1) < h(1, i), \tag{1}$$

$$\forall i \in \{2, \ldots, n\}: g_2(2, i+1) + g_1(2, i) > g_1(2, i+1) + g_2(1, i+1). \tag{2}$$

Now we assume that the vertical edge connects the nodes $(2, i)$ and $(1, i)$. If the node $(1, i)$ belongs to the top part, we have the situation of Figure 5b. The node $(2, i-1)$ belongs to the top part, too, since $2 \cdot (i-1) \geq 1 \cdot i$. By (1), it is easy to see that π_2^* is shorter than π_2, in contradiction to the assumption that $((2, i), (1, i))$ belongs to a shortest path. If the node $(2, i+1)$ belongs to the bottom part, we obtain by (2) that in Figure 5c the path π_3^* is shorter than π_3. Again $((2, i), (1, i))$ cannot belong to a shortest path. We conclude that the vertical edge appears in the middle part or in the top of the bottom part. Hence, the sum $\sum_{i=\lceil n^{1/2}/2 \rceil}^{n+1} h(2, i) \geq n^2/2 - n^{3/2}$ is a lower bound on the ROBDD size for an optimal variable ordering. For the symmetric variable ordering corresponding to π_4, the ROBDD size is $T_1(2,1) + \sum_{i=2}^{n+1} T_2(2, i) \leq h(2,1) + \sum_{i=2}^{n+1} h(2, i) = n^2/2 + O(n)$. Finally, the ROBDD

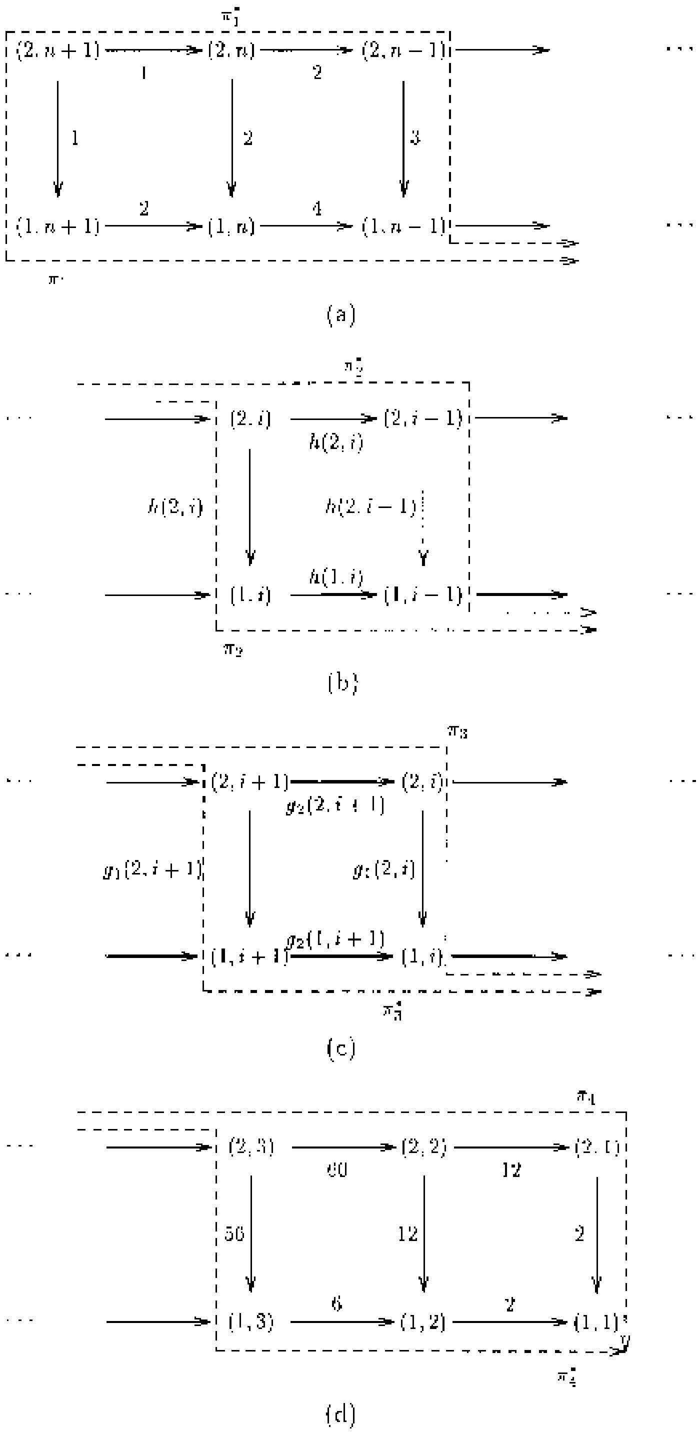

Fig. 5. (a), (b) The top part and (c), (d) the bottom part of the grid graph of a random function in PS$(1, n)$. The edge labels hold with probability $1 - e^{-\Omega(n^{1/2})}$.

size for the symmetric variable ordering corresponding to π_1 is $T_1(2, n+1) + \Sigma_{i=2}^{n+1} T_2(1, i) \le h(2, n+1) + \Sigma_{i=2}^{n+1} h(1, i) = n^2 + O(n)$. Altogether, the performance ratio converges to 1 as n goes to infinity. This implies Theorem 2.

7. PROOF OF THEOREM 3

First we prove Theorem 3 for the case of QOBDDs and discuss the generalization to ROBDDs later on. Let f be a random function chosen from $\mathrm{PS}^d(n-1)$. We are going to compare the edge labels of the paths corresponding to some symmetric variable ordering and some nonsymmetric variable ordering π in order to prove that we cannot save OBDD nodes by a nonsymmetric variable ordering. With probability $1 - o(1)$ in the top part a nonsymmetric variable ordering cannot be better than a symmetric variable ordering, but a nonsymmetric variable ordering may be better in the middle and in the bottom part. However, in this case, the top part of the OBDD is much larger for the nonsymmetric variable ordering, so we lose more OBDD nodes in the top part than we save in the middle and in the bottom part by using a nonsymmetric variable ordering.

First we remark that the grid graph can be partitioned into layers corresponding to the levels of OBDDs. At a node $(i_1, \dots, i_d)$, we have the situation that $n - i_j$ variables of the jth symmetry set are tested before. Altogether $dn - i_1 - \cdots - i_d$ variables are tested before. Hence, the labels $S(i_1, \dots, i_d)$, where $i_1 + \cdots + i_d = c$ for some constant c, describe the possible numbers of QOBDD nodes of the $(dn - c + 1)$th level of the QOBDD. We compare the estimates for the number of QOBDD nodes for symmetric and nonsymmetric variable orderings layerwise.

In order to simplify the notation, we only consider nodes $(i_1, \dots, i_d)$ of the grid graph, where $i_1 \le \cdots \le i_d$. If we consider a path running through some node $(i_1, \dots, i_d)$ not fulfilling this requirement, we reorder the components of $(i_1, \dots, i_d)$. Of course this leads to a different path and a different variable ordering, but since we get the same estimates for $S(i_1, \dots, i_d)$ and $S(i_{\tau(1)}, \dots, i_{\tau(d)})$ for each permutation τ, our estimates for the OBDD size do not change. In particular, we consider only one symmetric variable ordering, namely, the variable ordering corresponding to the path through the nodes

$$(n, \dots, n), (n-1, n, \dots, n), \dots, (1, n, \dots, n), \dots, (1, 1, n, \dots, n), \dots, (1, \dots, 1).$$

Note that the paths for symmetric variable orderings only run through those nodes $(i_1, \dots, i_d)$ where at most one component is different from 1 and n, because there may be at most one symmetry set from which at least one but not all variables are tested.

We choose $\varepsilon = 1/5$ and assume that for the function f, the events (E1) and (E5) occur. This holds with probability $1 - o(1)$. Let π be some nonsymmetric variable ordering.

Case 1. The path corresponding to π runs through the node $(1, \dots, 1, n)$. Then the path runs through all nodes $(1, \dots, 1, s)$, where $s \in \{2, \dots, n\}$, since all these nodes have the outdegree 1. Since π is nonsymmetric, there is at least one layer of the grid graph above the layer of the node $(1, \dots, 1, n)$, where the path for π runs through some node $(i'_1, \ldots, i'_d)$ for which at least two components are different from 1 and different from n. Otherwise, π would be symmetric. Let $(i_1, \ldots, i_d)$ be the node of the symmetric variable ordering on the same layer as $(i'_1, \ldots, i'_d)$. Then we have $i_1 + \cdots + i_d = i'_1 + \cdots + i'_d$. Furthermore, at most one component of $(i_1, \ldots, i_d)$ is different from 1 and different from n. Since d is a constant, both $(i_1, \ldots, i_d)$ and $(i'_1, \ldots, i'_d)$ belong to the top part. Then we have $S(i_1, \ldots, i_d) = h(i_1, \ldots, i_d)$ and $S(i'_1, \ldots, i'_d) = h(i'_1, \ldots, i'_d)$. The function $h(i_1, \ldots, i_d)$ takes its minimum value under the condition $i_1 + \cdots + i_d = c$ for some constant c iff at most one component of $(i_1, \ldots, i_d)$ is different from 1 and different from n. This can be proved in the following way. Since $h(i_1, \ldots, i_d)$ is equal to the product $(n+1-i_1) \cdot \cdots \cdot (n+1-i_d)$, the condition $i_1 + \cdots + i_d = c$ is equivalent to the condition that the sum of the factors of this product is constant. Hence, the task to minimize $h(i_1, \ldots, i_d)$ is the task to minimize a product under the condition that the sum of the factors is constant. We conclude that the QOBDD size for π is larger than the QOBDD size for the symmetric variable ordering.

Case 2. The path corresponding to π does not run through the node $(1, \ldots, 1, n)$. The proof for Case 1 shows that on the layers above the node $(1, \ldots, 1, n)$, a deviation from the path for the symmetric variable ordering leads to a larger QOBDD size. In the following, we assume that the QOBDD size on the levels corresponding to those layers is the same for π and for the symmetric variable ordering. Then we have to consider only the layers of the nodes $(1, \ldots, 1, s)$, where $s \in \{2, \ldots, n\}$. Let s^* be the largest number for which the path for π runs through the node $(1, \ldots, 1, s^*)$. This is the node where the paths for π and the symmetric variable ordering join. Afterward, the paths cannot split, since the nodes $(1, \ldots, 1, s)$, $s \geq 2$, have the outdegree 1. Hence, we have to consider only the layers of the nodes $(1, \ldots, 1, s)$, where $s \in \{s^* + 1, \ldots, n\}$. We are going to show that for all possible s^*, the variable ordering π leads to larger QOBDDs.

The QOBDD size for the layer of the node $(1, \ldots, 1, s)$ and the symmetric variable ordering is $S(1, \ldots, 1, s)$. For the variable ordering π we get a lower bound on the QOBDD size of the same level by taking the minimum of all $S(i_1, \ldots, i_d)$ of the same layer (i.e., $i_1 + \cdots + i_d = d - 1 + s$), where $S(1, \ldots, 1, s)$ is excluded. We consider the three parts of the grid graph separately. We are going to show that by a nonsymmetric variable ordering, we lose more OBDD nodes in the top part than we can save in the middle and in the bottom part.

Top Part. We consider the layers containing the nodes $(1, \ldots, 1, s)$, where $n \geq s \geq (2d + \varepsilon)\log n$. All nodes $(i_1, \ldots, i_d)$, where $i_1 + \cdots + i_d = d - 1 + s$, belong to the top part of the grid graph since $i_1 \cdot \cdots \cdot i_d \geq (2d + \varepsilon)\log n$. Hence, by (E1) for all $(i_1, \ldots, i_d)$, where $i_1 + \cdots + i_d = d - 1 + s$, it holds that $S(i_1, \ldots, i_d) = h(i_1, \ldots, i_d)$.

Then the number of QOBDD nodes on the level corresponding to $(1, \ldots, 1, s)$ and the symmetric variable ordering is $S(1, \ldots, 1, s) = h(1, \ldots, 1, s) = n^{d-1}(n - s + 1)$. In order to get a lower bound for the number of QOBDD nodes for the variable ordering π and the same level, we minimize $h(i_1, \ldots, i_d)$ under the conditions $i_1 + \cdots + i_d = d - 1 + s$ and $(i_1, \ldots, i_d) \neq (1, \ldots, 1, s)$. Again this is the task to minimize a product under the condition that the sum of the factors is constant. Since this product takes its minimum only for $(i_1, \ldots, i_d) = (1, \ldots, 1, s)$, we have to search for the second smallest value of this product, which it assumes for $(i_1, \ldots, i_d) =$

$(1,\ldots,1,2,s-1)$. Then we have $h(1,\ldots,1,2,s-1) = n^{d-2}(n-1)(n-s+2)$. The difference between the QOBDD size for π and symmetric variable orderings is at least $n^{d-2}(n-1)(n-s+2) - n^{d-1}(n-s+1) = n^{d-2}(s-2)$. Hence, if $s^* \geq (2d+\varepsilon)\log n$, then the QOBDD size for π is larger than for the symmetric variable ordering. We conclude $s^* < (2d+\varepsilon)\log n$. However, on the layers of the nodes $(1,\ldots,1,s)$, where $n \geq s \geq (2d+\varepsilon)\log n$, the number of QOBDD nodes for π is at least the sum of the number of QOBDD nodes for the symmetric variable ordering and L, where

$$L := \sum_{s=\lceil (2d+\varepsilon)\log n \rceil}^{n} n^{d-2}(s-2) = \Theta(n^d).$$

Hence, on the remaining levels of the QOBDD, at least L nodes have to be saved by using π instead of the symmetric variable ordering. If this is not possible, π is not optimal.

Middle Part. We consider the layers containing the nodes $(1,\ldots,1,s)$, where $(2d+\varepsilon)\log n > s > (d-\varepsilon)\log n$.

For the symmetric variable ordering we estimate $S(1,\ldots,1,s)$ by $h(1,\ldots,1,s) = n^{d-1}(n-s+1)$. For the lower bound on the QOBDD size for π, we consider all nodes $(i_1,\ldots,i_d)$, where $i_1 + \cdots + i_d = d-1+s$, except $(1,\ldots,1,s)$. Each such node may belong to the top part or to the middle part. If $(i_1,\ldots,i_d)$ belongs to the top part, then $S(i_1,\ldots,i_d) = h(i_1,\ldots,i_d) > h(1,\ldots,1,s)$. In the following, we get for $S(i_1,\ldots,i_d)$ for nodes $(i_1,\ldots,i_d)$ of the middle part, smaller estimates than $h(1,\ldots,1,s)$. Hence, it suffices to consider the nodes of the middle part. By (E5) we have $S(i_1,\ldots,i_d) \geq h(i_1,\ldots,i_d) - 4n$. For sufficiently large n and nodes $(i_1,\ldots,i_d)$ of the middle part, it holds that $h(i_1,\ldots,i_d) \geq n^d - d(2d+\varepsilon)n^{d-1}\log n$. This follows from the definition of $h(i_1,\ldots,i_d)$ by estimating each i_j by $(2d+\varepsilon)\log n$ and applying the binomial theorem afterward. Hence, the number of nodes that can be saved on the layer of $(1,\ldots,1,s)$ by using π instead of the symmetric variable ordering is bounded by

$$S(1,\ldots,1,s) - \bigl(h(i_1,\ldots,i_d) - 4n\bigr) \leq d(d+2\varepsilon)n^{d-1}\log n + 4n.$$

The number of such layers is bounded by $(2d+\varepsilon)\log n - (d-\varepsilon)\log n = (d+2\varepsilon)\log n$. Altogether, at most $O(n^{d-1}\log^2 n)$ nodes can be saved by using π. For sufficiently large n, these are less nodes than we lost in the top part. Hence, s^* has to be at most $(d-\varepsilon)\log n$. Otherwise π is not optimal.

Bottom Part. We consider the layers containing the nodes $(1,\ldots,1,s)$, where $s \leq (d-\varepsilon)\log n$. We have $S(1,\ldots,1,s) \leq g(1,\ldots,1,s) = 2^s$. We estimate the QOBDD size for these layers and π by 0. Hence, we save at most 2^s nodes. Then the number of saved QOBDD nodes for all layers in the bottom part is bounded by

$$\sum_{s=2}^{\lfloor (d-\varepsilon)\log n \rfloor} 2^s \leq 2n^{d-\varepsilon}.$$

Hence, if n is sufficiently large, by using π we lose more nodes in the top part than we can save in the middle part and in the bottom part. This implies that π is not optimal.

Finally, we discuss how to generalize the proof to ROBDDs. We assume that (E1), (E2), (E5), and (E6) occur. This is valid with probability $1 - o(1)$. Because of (E2) for nodes $(i_1, \ldots, i_d)$ of the top part and all $j \in \{1, \ldots, d\}$, where $i_j \geq 2$, it holds that $S(i_1, \ldots, i_d) = T_j(i_1, \ldots, i_d)$. Hence, the proof does not change for Case 1 and the top part of Case 2. Now we consider the middle part, i.e., the layers containing the nodes $(1, \ldots, 1, s)$, where $(2d + \varepsilon)\log n > s > (d - \varepsilon)\log n$. Again we estimate the ROBDD size of each such layer by $h(1, \ldots, 1, s)$. For π, we may save more nodes than in the case of QOBDDs because for ROBDDs we do not count the blocks that are constant in a particular direction. By (E5) and (E6) we have for all j, where $i_j \geq 2$, that $T_j(i_1, \ldots, i_d) \geq h(i_1, \ldots, i_d) - 4n - 2n^{d/2+\varepsilon/2}$ holds. Hence the number of nodes saved by using π is bounded by $S(1, \ldots, 1, s) - (h(i_1, \ldots, i_d) - 4n - 2n^{d/2+\varepsilon/2}) = d(d + 2\varepsilon)n^{d-1}\log n + 4n + 2n^{d/2+\varepsilon/2}$. The number of layers is bounded by $(d + 2\varepsilon)\log n$. Hence, we save altogether at most $O(n^{d-1}\log^2 n)$, if $d \geq 3$, or $O(n^{1+\varepsilon/2}\log n)$ nodes, if $d = 2$. These are asymptotically less than $L = \Theta(n^d)$ nodes.

For the bottom part we estimate again the number of ROBDD nodes for the symmetric variable ordering and the layer of $(1, \ldots, 1, s)$ by $g(1, \ldots, 1, s) = 2^s$. Then the proof does not change. Altogether, we have proved Theorem 3 for ROBDDs and QOBDDs.

ACKNOWLEDGMENT

I thank Ingo Wegener for many hints and fruitful discussions.

REFERENCES

[1] B. Bollig and I. Wegener, Improving the variable ordering of OBDDs is *NP*-complete, *IEEE Trans. Comput.* **45**, 993–1002 (1996).

[2] R. E. Bryant, Graph-based algorithms for Boolean function manipulation, *IEEE Trans. Comput.* **35**, 677–691 (1986).

[3] R. E. Bryant, Symbolic Boolean manipulation with ordered binary-decision diagrams, *ACM Comput. Surveys* **24**, 293–318 (1992).

[4] K. M. Butler, D. E. Ross, R. Kapur, and M. R. Mercer, Heuristics to compute variable orderings for efficient manipulation of ordered binary decision diagrams, in *Proceedings of the 28th Design Automation Conference*, 1991, pp. 417–420.

[5] M. Fujita, H. Fujisawa, and N. Kawato, Evaluation and improvements of Boolean comparison method based on binary decision diagrams, in *Proceedings of the International Conference on Computer-Aided Design*, 1988, pp. 2–5.

[6] S.-W. Jeong, T.-S. Kim, and F. Somenzi, An efficient method for optimal BDD ordering computation, in *Proceedings of the International Conference on VLSI and CAD (ICVC)*, 1993, pp. 252–256.

[7] B.-G. Kim and D. L. Dietmeyer, Multilevel logic synthesis of symmetric switching functions, *IEEE Trans. Computer-Aided Design* **10**, 436–446 (1991).

[8] V. F. Kolchin, B. A. Sevast'yanov, and V. P. Christyakov, *Random Allocations*, Wiley, New York, 1978.

[9] M. Löbbing and I. Wegener, The number of knight's tours equals 33,439,123,484,294—Counting with binary decision diagrams, *Electron. J. Combin.* **3**, #R5 (1996).

[10] S. Malik, A. R. Wang, R. K. Brayton, and A. Sangiovanni-Vincentelli, Logic verification using binary decision diagrams in a logic synthesis environment, *Proceedings of the International Conference on Computer-Aided Design*, 1988, pp. 6–9.

[11] D. Möller, J. Mohnke, and M. Weber, Detection of symmetry of Boolean functions represented by ROBDDs, *Proceedings of the International Conference on Computer-Aided Design*, 1993, pp. 680–684.

[12] D. Möller, P. Molitor, and R. Drechsler, Symmetry based variable ordering for ROBDDs, in *Proceedings of the IFIP Workshop on Logic and Architecture Synthesis*, 1994, pp. 47–53.

[13] S. Panda and F. Somenzi, Who are the variables in your neighborhood, *Proceedings of the International Conference on Computer-Aided Design*, 1995, pp. 74–77.

[14] S. Panda, F. Somenzi, and B. F. Plessier, Symmetry detection and dynamic variable ordering of decision diagrams, *Proceedings of the International Conference on Computer-Aided Design*, 1994, pp. 628–631.

[15] R. Rudell, Dynamic variable ordering for ordered binary decision diagrams, *Proceedings of the International Conference on Computer-Aided Design*, 1993, pp. 42–47.

[16] D. Sieling, The nonapproximability of OBDD minimization, Technical Report 663, Universität Dortmund, 1998.

[17] D. Sieling and I. Wegener, *NC*-algorithms for operations on binary decision diagrams, *Parallel Process. Lett.* **3**, 3–12 (1993).

[18] D. Sieling and I. Wegener, On the representation of partially symmetric Boolean functions by ordered multiple valued decision diagrams, Technical Report 627, Universität Dortmund, 1996; *Multiple Valued Logic*, Special Issue on Decision Diagrams, to appear.

[19] I. Wegener, Optimal decision trees and one-time-only branching programs for symmetric Boolean functions, *Inform. Control* **62**, 129–143 (1984).

[20] I. Wegener, Efficient data structures for Boolean functions, *Discrete Math.* **136**, 347–372 (1994).

[21] I. Wegener, The size of reduced OBDD's and optimal read-once branching programs for almost all Boolean functions, *IEEE Trans. Comput.* **43**, 1262–1269 (1994).