On the Relation Between Disjunctive Decomposition and ROBDD Variable Ordering*

E. V. Dubrova D.M. Miller J. C. Muzio
VLSI Design and Test Group
Department of Computer Science
University of Victoria, P.O.Box 3055
Victoria, B.C., Canada, V8W 3P6
{elena, mmiller, jmuzio}@csr.uvic.ca

Abstract

The relation between disjunctive decomposition of functions and a variable ordering minimizing the size of a ROBDD, is considered. We show that a best ordering for a function with a disjunctive decomposition cannot always be directly determined from the best orderings for the component functions. We also demonstrate that keeping the variables from a bound set of the function adjacent does not always guarantee obtaining the ROBDD for the function with a minimal number of nodes.

1. Introduction

This paper considers the problem of finding a variable ordering which minimizes the size of a Reduced Ordered Binary Decision Diagram (ROBDD) for functions possessing disjunctive decompositions of the type

$$f(X) = g(h(Y), Z)$$

with Y and Z being sets of variables forming a partition of the set $X = \{x_1, x_2, \ldots, x_n\}$.

A Reduced Ordered Binary Decision Diagram (ROBDD) is a graphical data structure for the efficient representation of a Boolean function. Functions are represented by directed, acyclic graphs, which are built for some chosen ordering of the function variables. Although a function may require, in the worst case, a graph of size exponential in the number of variables, many practical functions have a representation which is linear in the number of variables.

For functions with disjunctive decompositions, storage can be saved by expressing them as a composition $f(X) = g(h(Y), Z)$ of two functions g and h, and storing the ROBDDs of g and h [1]. When needed, the ROBDD for f can be expanded in a straightforward fashion by replacing the composition variable in the ROBDD of g with the graph for h, and then reducing the resulting diagram. If C_n is an upper bound on the number of nodes in a ROBDD for function of n variables, then the total number of nodes in ROBDDS for g and h is bounded above by $C_p + C_{1+n-p}$, where p is the number of variables in h. Because the bound C_n increases nearly exponentially with n [8], the discovery of any nontrivial decomposition of the form $f(X) = g(h(Y), Z)$ might greatly save storage space for f.

Normally, the size of the ROBDD varies for different variable orderings and, for some functions, it is highly sensitive to the ordering. Finding a best ordering that minimizes the size of the graph requires, in the worst case, time exponential in the number of variables. Therefore, computing the best orderings for two functions of n_1 and n_2 variables respectively, is usually much faster than computing a best ordering for one function of $n_1 + n_2$ variables. Thus, a natural question to ask for functions with disjunctive decompositions is whether a best ordering for $f = g(h(Y), Z)$ can always be determined from the best orderings for h and g. Intuitively, one is tempted to say that it should be so, but in [5] we gave a counterexample showing that for $n \geq 5$ an ordering generated from the best orderings of g and h is sometimes not a best one for f.

In this paper we analyze the result from [5] in more detail. We also study whether one can use a bound set of a function as a criterion for grouping the variables to obtain a best ordering for the ROBDD for this function. It seems right to suggest that the variables from a bound set should be kept adjacent to guarantee obtaining the ROBDD with a minimal number of nodes. One wouldn't expect that dispersing the variables from a bound set, can lead to the reduction of the number of nodes in the ROBDD. However, we show a

*Supported in part by Research Grant No. 5711 from the Natural Sciences and Engineering Research Council of Canada and by an equipment loan from the Canadian Microelectronic Corporation.

0-7803-3905-3/97/$10.00

counterexample to this intuition.

The paper is organized as follows. In Section 2 background on disjunctive decomposition and ROBDD's orderings is given. Section 3 considers the problem of determining the best ordering for a decomposable function $f = g(h(Y), Z)$ from the best orderings for g and h. Section 4 shows that grouping the variables from a bound set doesn't always guarantee obtaining the ROBDD with a minimal number of nodes. The paper concludes with suggestions for work following from this research.

2 Background

Let $f(x_1, x_2, \ldots, x_n)$ be a completely specified Boolean function of type $f : B^n \to B$ on $B = \{0, 1\}$. We denote by X the set of the variables of f, i.e. $X = \{x_1, x_2, \ldots, x_n\}$.

Let Y denote a proper subset of X, and let $Z = X - Y$. The operation *functional substitution* of a function h into a variable of another function g is defined if $h : B^{|Y|} \to B$, $g : B \times B^{|Z|} \to B$, resulting in the function $f : B^{|Y|} \times B^{|Z|} \to B$ given by

$$f(X) = g(h(Y), Z) \qquad (1)$$

Conversely, (1) is a *decomposition* of f for a suitably chosen set of variables Y. Any set Y of variables for which a representation $f(X) = g(h(Y), Z)$ is possible is called a *bound* set for f. For Boolean functions, bound sets have been first studied by Ashenhurst [2]. A decomposition (1) always exists for Y given by any singleton set $\{x_i\}$ or the all-set $\{x_1, x_2, \ldots, x_n\}$. Such sets are called *trivial* bound sets. A function f that has only trivial bound sets is called *undecomposable*.

An *ordering* of the variables in a ROBDD for f is a vector, describing the variables in order from top to bottom of the ROBDD. A *best ordering* is the ordering resulting in the ROBDD with a minimal number of nodes.

The operation *ordering substitution* of an ordering $< Y >$ into a variable h of another ordering $< Z_1, h, Z_2 >$ is defined if $Z_1 \cup Z_2 = Z, h \notin Z$ and $Z_1 \cap Z_2 = \emptyset$, resulting in the ordering $< X >$ given by

$$< X > = < Z_1, < Y >, Z_2 > .$$

Note, that without any confusion we are using h to also denote the substituted variable of g.

3 Obtaining a best ordering for f from best orderings for g and h.

We are interested as to whether the set of best orderings for a function with a disjunctive decomposition $f(X) = g(h(Y), Z)$ can always be composed from the best orderings of g and h, i.e. be calculated by performing ordering substitution on the sets of best orderings for h and g. In more formal terms, this question can be expressed as follows.

Let S_1 be the set of all non-degenerate (i.e. depending on all their input variables) functions of n variables or less. Let S_2 be the set of all sets, which are best orderings of the functions from S_1. Let $\alpha : S_1 \to S_2$, be defined as the mapping assigning to any function $f \in S_1$ the set of best orderings for f from S_2. If $\circ$ denotes functional substitution, and $\bullet$ denotes ordering substitution, then, we check the existence of a homomorphism between two structures $(S_1, \circ)$ and $(S_2, \bullet)$.

These two structures are homomorphic if, and only if, there exists a mapping $\alpha : S_1 \to S_2$ assigning to any function $f \in S_1$ the set of best orderings for f from S_2, such that

$$\alpha(g \circ h) = \alpha(g) \bullet \alpha(h)$$

for all $g, h \in S_1$ for which the operation $\circ$ is defined. Here $\alpha(g \circ h)$ is the set of all best orderings for $f(X) = g(h(Y), Z)$, and $\alpha(g) \bullet \alpha(h)$ is the set obtained after performing ordering substitution on the sets of best orderings for h and g.

The following theorem shows, however, that this is not the case for $n \geq 5$.

Theorem 1 *[5] Let $\alpha : S_1 \to S_2$ be the mapping assigning to any function $f \in S_1$ the set of best orderings for f from S_2. Then for $n \geq 5$ α is not a homomorphism between $(S_1, \circ)$ and $(S_2, \bullet)$.*

Proof: By example. Consider the following function of 5 variables:

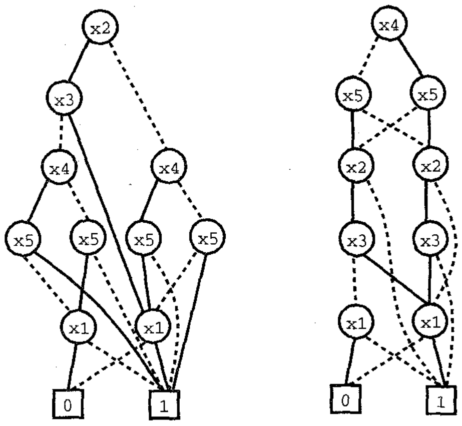
$$f(x_1, \ldots, x_5) = x_1(x_4 \oplus x_5)' + x_2'(x_4 \oplus x_5) + x_1 x_3 + x_1' x_2 x_3'$$

It can be decomposed as $f = g(h(x_4, x_5), x_1, x_2, x_3)$, where

$$g = x_1 h' + x_2' h + x_1 x_3 + x_1' x_2 x_3' \quad \text{and} \quad h = x_4 \oplus x_5$$

The ordering $< x_2, x_3, h, x_1 >$ is the only best ordering for g, resulting in a ROBDD with 8 nodes. Therefore $\alpha(g) = \{< x_2, x_3, h, x_1 >\}$. Since h is totally symmetric, all its orderings give ROBDDs with the same number of nodes, and therefore $\alpha(h) = \{< x_4, x_5 >, < x_5, x_4 >\}$. So $\alpha(g) \bullet \alpha(h) = \{< x_2, x_3, x_4, x_5, x_1 >, < x_2, x_3, x_5, x_4, x_1 >\}$. Both of these two orderings result in ROBDDs for f with 12 nodes. For example, the ROBDD for the ordering $< x_2, x_3, x_4, x_5, x_1 >$ is shown on Figure 1(a).

However, there exist orderings for f yielding a ROBDD with 11 nodes, as for example the ordering $< x_4, x_5, x_2, x_3, x_1 >$ (Figure 1(b)), and therefore $\alpha(g \circ h) \neq \alpha(g) \bullet \alpha(h)$ for $n = 5$.



(a) (b)

Figure 1. ROBDDs for f for the orderings (a) $< x_2, x_3, x_4, x_5, x_1 >$ and (b) $< x_4, x_5, x_2, x_3, x_1 >$.

The phenomenon demonstrated by the above example holds for any h, as long as h is a function of two or more variables, because in the ROBDD for g for the ordering $< x_2, x_3, h, x_1 >$ the variable h is represented by two nodes, while in the ROBDD for the ordering $< h, x_2, x_3, x_1 >$ the variable h is represented by just one node. Thus the theorem holds for $n \geq 5$.

□

The above example, and hence the theorem, are also applicable for the case where output edge negation [9] is allowed in the ROBDD.

Theorem 1 gives a negative answer to the question whether a best ordering for a function of 5 or more variables with a disjunctive decomposition $f(X) = g(h(Y), Z)$ can always be determined from the best orderings of h and g. We believe that for $n < 5$ the answer is positive.

The example, given in the proof, shows that for $n \geq 5$ sometimes an ordering generated from the best orderings of g and h is not a best one for f. Furthermore, it demonstrates that it is possible that *none* of the orderings generated this way are best for f. Such examples, however, are quite rare, and are hard to find. In most of the cases, the ordering obtained by performing ordering substitution on the sets of best orderings for h and g is a best one. In the rare cases it is not a best, it is still very close to a best. Furthermore if f is a practical function of a large number of variables, one wouldn't be able to compute a best ordering for f anyway, since the existing exact algorithms for finding best orderings are feasible only for functions of a small number of variables. Therefore, using a decomposition for partitioning the problem of finding a best variable ordering into several smaller ones is valuable and useful, in spite of the counterexample shown.

4. Using a bound set as a criterion for finding a best ordering.

Since the problem of computing a best variable ordering is NP-complete [6], the exact algorithms for its solution are feasible only for functions of a small number of variables. A number of heuristic procedures have been developed, using various strategies to produce a "good" ordering within a reasonable time.

A key to developing an efficient heuristic procedure for computing a good variable ordering lies in formulating a dependable criterion for grouping the variables. For example, it was empirically observed in [7], that symmetric variables tend to be adjacent in the best ordering for ROBDDs without complemented edges. Since then, keeping symmetric variables together has been considered a good criterion and a number of heuristic procedures for computing a variable orderings were developed, based on this criterion, including whose in [10] and [11]. However, later a counterexample was found in [11], showing a function for which no symmetric order is optimal.

Keeping variables from a bound set of a function f adjacent seems to be another intuitive criterion for grouping variables to guarantee obtaining the ROBDD with a minimal number of nodes. One wouldn't expect that dispersing the variables from a bound set apart, can lead to a reduction of the number of nodes in the ROBDD. Below, however, we show an example for one such case.

Let Y be a proper subset of X. We say that in the ordering $< X >$ the variables from Y are *adjacent*, if $< X >$ can be represented as $< X > = < Z_1, < Y >, Z_2 >$, where $Z_1 \cup Z_2 = Z, Z_1 \cap Z_2 = \emptyset$ and $Z = X - Y$. Otherwise we say that in the ordering $< X >$ the variables from Y are *dispersed*.

Theorem 2 *There exists a decomposable function $f = g(h(Y), Z)$ such that the number of nodes in its ROBDD for an ordering with the variables from Y dispersed is smaller than the number of nodes in its ROBDD for an ordering with the variables from Y adjacent.*

Proof: By example. Consider the following function of 6 variables $f(x_1, \ldots, x_6)$:

$$f = x_1'(x_3x_4' + x_3'(x_5' + x_6')) + (x_1 + x_2)(x_3x_4 + x_3'x_5x_6)$$

It can be decomposed as $f = g(h(x_3, x_4, x_5, x_6), x_1, x_2)$, where

$$g = x_1'h + (x_1 + x_2)h' \quad \text{and} \quad h = x_3x_4' + x_3'(x_5' + x_6')$$

The ROBDD for f has 12 nodes for the ordering $< x_1x_3x_5x_6x_4x_2 >$, which is an ordering where the

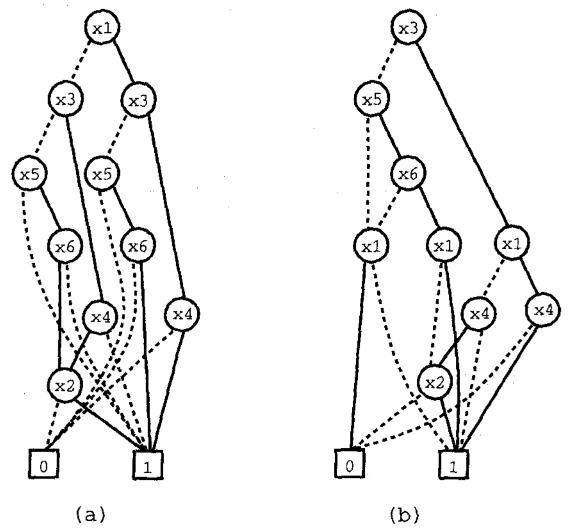


Figure 2. ROBDDs for f for the orderings (a) $< x_1, x_3, x_5, x_6, x_4, x_2 >$ and (b) $< x_3, x_5, x_6, x_1, x_4, x_2 >$.

variables from the bound set $\{x_3, x_4, x_5, x_6\}$ are adjacent (Figure 2(a)), and 11 nodes for the ordering $< x_3, x_5, x_6, x_1, x_4, x_2 >$, which is an ordering where the variables from the bound set are dispersed. (Figure 2(b)).

□

The above example, and hence the theorem, are also applicable for the case where output edge negation is allowed in the ROBDD. In this case the ROBDD for f has 11 nodes for the ordering $< x_1, x_3, x_5, x_6, x_4, x_2 >$, and 10 nodes for the ordering $< x_3, x_5, x_6, x_1, x_4, x_2 >$.

Theorem 2 shows that keeping the variables from a bound set adjacent doesn't always guarantee obtaining the ROBDD with a minimal number of nodes. Such cases, however, are extremely rare and, on practice, it is almost always a good idea to keep the variables from the bound set together.

5. Conclusion

This paper considers the problem of finding a variable ordering which minimizes the size of a Reduced Ordered Binary Decision Diagram for functions possessing disjunctive decompositions.

The example, given in the proof of the Theorem 1, shows that for $n \geq 5$ sometimes an ordering generated by performing ordering substitution on the sets of best orderings for h and g is not a best one for f. The example, given in the proof of the Theorem 2, shows that keeping the variables from a bound set adjacent doesn't always guarantee obtaining the ROBDD with a minimal number of nodes. Such examples, however, are quite rare. Their existence doesn't diminish the practical value of applying decomposition techniques to solving the ROBDD variable ordering problem, but should be noted as a possibility.

References

[1] S.B. Akers, Binary Decision Diagrams, *IEEE Transactions on Computers* **C-27** No. 6 (1978), 509-516.

[2] R. L. Ashenhurst. The decomposition of switching functions. *Proc. Int. Symp. Theory of Switching*, Part I, Ann. Comput. Lab. Harvard Univ. **29** (1959), 74–116.

[3] G. Birkhoff, S. MacLane, Brief Survey of Modern Algebra, 4th ed., New York, Macmillan, 1977.

[4] R.E. Bryant, Graph-Based Algorithm for Boolean Function Manipulation, *IEEE Transactions on Computers* **C-35** No. 8 (1986), 677-691.

[5] E.V. Dubrova, D.M. Miller, J.C. Muzio, On the Best ROBDD Variable Ordering for Functions with Disjunctive Decompositions, *IEE Journal of Electronic Letters*, to appear.

[6] S.J. Friedman, K.J. Supowit, Finding the Optimal Variable Ordering for Binary Decision Diagrams, in *Proc. 24th ACM/IEEE Design Automation Conf.* (1987), 348-355.

[7] S.-W. Jeong, B. Plessier, G.D. Hatchel, F. Somenzi, Variable Ordering and Selection for SSM traversal, in *Proceedings of the IEEE Int. Conf. on Computer Aided Design* (1991), 476-479.

[8] M. Heap, On the Exact Ordered Binary Decision Diagram Size of Totally Symmetric Functions, *Journal of Electronic Testing: Theory and Applications* **4** No. 2 (1993), 191-195.

[9] S. Minato, N. Ishiura, S. Yajima, Shared binary decision diagrams with attributed edges for efficient Boolean function manipulation, in *Proceedings of 27th ACM/IEEE Design Automation Conf.* (1990), 52-57.

[10] D. Möller, P. Molitor, R. Drechsler, Symmetry based variable ordering for ROBDDs, in *IFIP Workshop on Logic and Architecture Synthesis* (1994).

[11] S. Panda, F. Somenzi, Who are the Variables in Your Neighborhood, in *Proceedings of IEE/ACM Workshop on Logic Synthesis* (1995), 1-10.