Variable Reordering for Shared Binary Decision Diagrams Using Output Probabilities*

M. A. Thornton, J. P. Williams
Department of Computer Engineering
University of Arkansas, Fayetteville, Arkansas, USA
{mat1,jpw}@engr.uark.edu

R. Drechsler, N. Drechsler
Institute of Computer Science
Albert-Ludwigs University, 79110 Freiburg, Germany
drechsle@informatik.uni-freiburg.de

Abstract

The Shared Binary Decision Diagram (SBDD) with negative edge attributes can represent many functions in a compact form if a proper variable ordering is used. In this work we describe a technique for reordering the variables in an SBDD to reduce the size of the data structure. We use a heuristic to formulate a technique for the reordering problem based on probability metrics.

1 Introduction

Many approaches for BDD variable reordering have been investigated. Techniques that depend on variable interrelationships such as symmetry have been applied to both variable ordering and reordering approaches [2, 4, 7, 5, 6]. Recently, it has been observed that the robustness of OBDD algorithms is a very important aspect [3], but the dynamic variable ordering algorithms above largely depend on the chosen starting point, since they are mainly greedy heuristics.

In this paper we describe a stable variable ordering heuristic that makes use of properties of the function only, but is not dependent on the initial representation. We utilize a method that was motivated by the behavior of symmetric variables in "best-known" SBDD orderings. Since we can compute output probability values efficiently on SBDDs, we use the quantities to indicate the possible existence of a symmetry relation.

2 Heuristic Description and Motivation

Many researchers have utilized heuristics that measure the relationship between primary inputs. Techniques that exploit relationships between dependent variables continue to be popular particularly since it is noted that the application of sifting [7] can lead to orderings where symmetric variables tend to cluster together [6].

The determination of symmetry among variables is not a computationally efficient process to undertake by pure manipulation of SBDD structures. Therefore an alternative method based on circuit output probabilities is formulated here to utilize symmetry indicators for the purpose of reordering. Theorem 1 forms the basis of our technique.

Theorem 1 $\wp\{f \oplus x_i\} = \wp\{f \oplus x_j\}$ *if* $x_i, x_j \in S$ *where* S *is the set of all independent variables that support* f *and* $x_i \leftrightarrow x_j$ *(that is,* x_i *and* x_j *are symmetric in* f*).*

During the formulation of this technique, we generated plots of $\wp\{f \oplus x_i\}$ versus the dependent variable index i for the best known orderings of several benchmark circuits in SBDD form. Due to past heuristics based on symmetric variables, we expected to see probability values with similar magnitudes clustered together. The resultant plots illustrated an unexpected trend of periodicity. This periodic trend suggests that dependent variables with the same $\wp\{f \oplus x_i\}$ values tend to position themselves as far apart from one another as possible. Figure 1 illustrates the behavior using the best known ordering for $c432$.

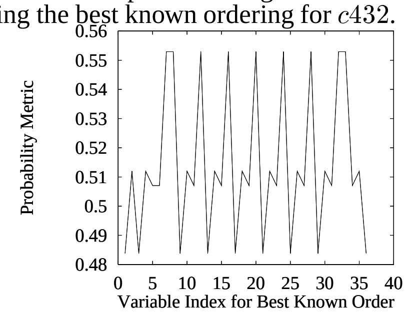

Figure 1. Behavior of $\wp\{f \oplus x_i\}$, Output $370gat$

In order to exploit the characteristic of periodicity, an algorithm was developed that orders the variables such that the corresponding $\wp\{f \oplus x_i\}$ values are as periodic as possible. The technique involves the following steps:

1. Choose an output about which to compute n values of $\wp\{f \oplus x_i\}$
2. Compute all the probability values
3. Determine histogram bin sizes and widths
4. Form a histogram of the $\wp\{f \oplus x_i\}$ values
5. Starting from the bin with the lowest (or highest) probability value, choose a value
6. Move to the next adjacent bin (in a circular fashion) and choose a value
7. When all values have been removed from the bin, build a list of dependent variables in the same order in which the probability values were chosen
8. Reorder the initial SBDD according to the list just generated
9. Perform a modified sifting routine, we refer to as "bin-sifting"

Figure 2 contains a plot of the $\wp\{f \oplus x_i\}$ values versus the index of x_i after application of the variable reordering algorithm. This data corresponds to the same output as was used in Figure 1, $370gat$. Clearly, periodicity of the probability values has been enforced.

2.1 Example of the Method

To illustrate the technique described above, a small example is shown. We have chosen the "toy benchmark" c17 for this purpose.

Initially, the ROBDD exists with a variable order X_2, X_3, X_6, X_7 and is shown graphically in Figure 3. The first step of the process is to compute the output probabilities, $\wp\{f \oplus x_i\} \forall i$. The computed output probabilities for the

*This project was supported by the NSF under grant MIP-9633085.

Table 1. Experimental Results for $ISCAS85$

benchmark		sift		bin-sift		prob. meth.		best-known sizes		
Circuit	Org. Size	Size	Time	Size	Time	Size	Time	SA	Best Sift	EA
c432	31172	20891	28	20891	216	1119	13	1087	1210	1064
c499	53866	38196	84	36234	156	34432	245	25866	25866	25866
c880	10348	5089	9	4603	20	4603	26	4053	4083	4053
1355	53866	38186	88	36234	159	34432	280	25866	25866	25866
c1908	13934	8175	11	7762	23	6170	48	5526	5708	5526
c3540	72858	41918	73	37644	1421	38096	257	23828	23828	23828
c5315	747662	4078	225	-	-	3447	4108	-	2104	1719

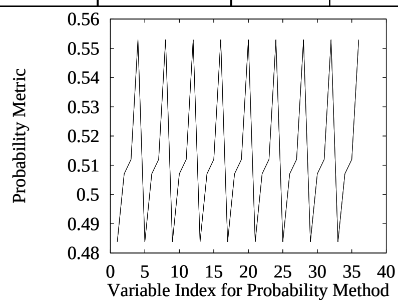

Figure 2. Probability Based Ordering

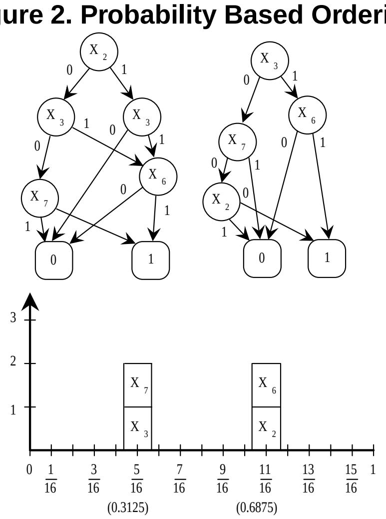

Figure 3. ROBDDs and Histogram

functions $f \oplus x_i$ are: $\wp\{f \oplus X_2\} = 0.6875$, $\wp\{f \oplus X_3\} = 0.3125$, $\wp\{f \oplus X_6\} = 0.6875$ and $\wp\{f \oplus X_7\} = 0.3125$.

A histogram is formed using the probability metrics. Figure 3 also contains the histogram. A new variable order is obtained by visiting the histogram bins and removing a variable in a circular fashion, thus ensuring periodicity of the $\wp\{f \oplus X_i\}$ values. The new order obtained from the histogram is X_7, X_6, X_3, X_2.

The last step of the reordering technique involves the application of the bin-sifting routine. Since the example contains two histogram bins, the bin-sifting approach allows for exchanging the locations of $\{X_3, X_7\}$ and $\{X_6, X_2\}$ only. The resulting variable order for the example function is found to be X_3, X_6, X_7, X_2 and ROBDD is also shown in Figure 3.

3 Experimental Results

We have computed orderings for several benchmark circuits using the techniques described above. We have compared these to sifting alone which yields reordering improvements in a small amount of CPU time, and best known orderings which were generally computed using techniques such as *Simulated Annealing* (SA) or *Evolutionary Algorithms* (EAs) that yield very good minimized SBDDs but can require larger amounts of computation time.

Table 2. Experimental Results for PLAs

circuit		sifting		prob. method	
Name	Org. Size	Size	Time	Size	Time
5xp1	74	51	< 1	42	< 1
alu4	1197	800	< 1	639	1
bw	108	103	< 1	99	< 1
duke2	973	394	< 1	339	1
misex1	41	37	< 1	37	< 1
misex2	136	89	< 1	84	< 1
misex3	1301	704	< 1	683	5
sao2	155	94	< 1	87	< 1
misex3c	828	504	< 1	472	1
clip	226	87	< 1	87	< 1
e64	1441	231	1	231	2
apex1	28336	1356	3	1333	58
apex4	928	895	< 1	889	1
apex5	2679	1130	< 1	1130	1

4 Conclusion

A variable reordering method for SBDDs with negative edge attributes has been presented and described. Experimental results indicate that runtimes are roughly equivalent to those of sifting, however smaller SBDDs generally result. The method differs significantly from those previously published in the way that function properties are used. This technique yields a resultant SBDD that is independent of the initial structure and is thus very robust.

The quality of our results has been shown by a comparison to an EA. The EA algorithm was able to further improve the best known SBDD sizes for many functions.

References

[1] B. Bollig, M. Löbbing, and I. Wegener. Simulated Annealing to Improve Variable Orderings for OBDDs. *Proceedings of the IEEE/ACM Workshop on Logic Synthesis*, 1995.

[2] M. Fujita, Y. Matsunaga, and T. Kakuda. On Variable Ordering of Binary Decision Diagrams for Applications of Multi-level Synthesis. *Proceedings of the European Conference on Design Automation*, pages 50–54, 1991.

[3] J. E. Harlow III and F. Brglez. Characterization of BDD Algorithms Using Circuit Mutant Equivalence Classes. *Proceedings of ICCAD*, pages 646-652, 1998.

[4] S. Minato. Minimum-Width Method of Variable Ordering for Binary Decision Diagrams. *IEICE Transactions on Fundamentals*, vol. E75-A, no. 3:392–399, 1992.

[5] D. Möller, P. Molitor, and R. Drechsler. Symmetry Based Variable Ordering for ROBDDs. *IFIP Workshop on Logic and Architecture Synthesis*, pages 47–53, 1994.

[6] S. Panda and F. Somenzi. Who are the Variables in Your Neighborhood. *Proceedings of ICCAD*, pages 74-77, 1995.

[7] R. Rudell. Ordering for Ordered Binary Decision Diagrams. *Proceedings of the International Conference on Computer Aided Design*, pages 42–47, November 1993.