Modified Haar Transform Calculation Using Digital Circuit Output Probabilities *

M. A. Thornton
Department of Computer Systems Engineering
University of Arkansas
Fayetteville, AR 72701-1201

Abstract

A method for the computation of Haar spectral coefficients is described using the OBDD representation of a function. An algebraic relationship between the circuit output probabilities and the Haar spectral coefficients is derived and used. The circuit output probabilities are computed by applying graph algorithms to the OBDDs. A fundamental dependence between the spectral coefficients and $n+1$ simple Boolean equivalence relationships is noted.

1 Introduction

The various forms of spectra of a Boolean function provide a unique means of function definition and each individual spectral coefficient provides information regarding the behavior of the function over various inputs. In the past, various spectra of Boolean functions have been used for analysis, synthesis, and verification tasks however the computational complexity associated with these computations limited widespread usage.

More recently, developments in the use of Decision Diagrams for Boolean function representation and their subsequent usage in the computation of spectral coefficients has renewed interest in spectral methods. Several techniques for computing spectral coefficients based on DDs have been developed [2] [10] [3] [11]. Here we utilize the concept of circuit output probabilities as a set of quantities that allow the modified Haar spectral coefficients to be calculated using simple algebraic equations.

Circuit output probabilities [7] were first proposed for the analysis of the effectiveness of random testing. Recently, these values have been reexamined due to their close relationship with 'signal switching probabilities', or 'switching activity factors' which are used extensively in the development and analysis of low power circuitry design and analysis methods [4] [8] [13].

The computation of the modified Haar transform [1] [5] coefficients is more complex than the probability based methods for computing Walsh and Reed-Muller spectral values [11] [12]. The increase in complexity is due to the fact that an independent constituent function no longer holds. In this case, the analogy to the constituent function used in the previous work is actually a sub-function of the function to be transformed. Since we are dealing with probabilities, a degree of independence is lost. For this reason, a single algebraic equation in terms of a constituent function can not be formulated as is shown in Section 3.

An alternative method for the computation of the Haar spectrum using BDDs has also been developed [3]. This technique is based upon the notion of 'extended literals' and the spectral coefficients are computed through the identification of particular paths contained within the BDD representing 'input combinations'. This approach differs from the technique described in this paper where we focus on the interaction between digital circuit output probabilities and particular modified Haar spectral coefficients.

2 Output Probability Computations

The output probability expression for a Boolean function is a real-valued algebraic equation that specifies the probability that the function is valued at logic-'1' given the probability distributions of each of the dependent variables. Therefore, the probability space consists of 2^n experiments where n is the number of dependent Boolean variables. If it is assumed that the function is fully specified, and that each input is equally likely to be 0 or 1, all probabilities for func-

*This research was supported by the National Science Foundation under contract MIP-963 3085.

tion variables may be set to $\frac{1}{2}$. The resulting circuit output probability can be viewed as the percentage of minterms that cause the function to evaluate to logic-'1'.

The output probabilities were first proposed by Parker and McCluskey [7] where they were used to evaluate the effectiveness of random testing for combinational logic circuits. Recently, interest has been renewed in these quantities since they can be used to form estimates of switching activity factors. Switching activity factors are very useful in the prediction of overall power dissipation for CMOS circuitry and are thus a quantity to be minimized for low power design [4] [8] [13].

Any of the methods mentioned in [6] [7] [11] may be used to determine the percentage of minterms that cause a function, f, to evaluate to logic-'1'. This quantity is denoted here as $\wp\{f\}$. As an example, consider the function defined by the Boolean expression in Equation 1.

$$f(x) = x_1x_3\overline{x}_6 + x_1\overline{x}_3x_4\overline{x}_6 + x_1\overline{x}_3\overline{x}_4\overline{x}_5 + \overline{x}_1x_2x_4\overline{x}_6 + \overline{x}_1x_2\overline{x}_4\overline{x}_5 + x_1\overline{x}_2\overline{x}_5 \quad (1)$$

Application of any of the techniques referenced above yield the result, $\wp\{f\} = \frac{5}{8}$. Thus, 62.5 % of the possible 2^6 minterms will cause f to evaluate to a logic-'1' value.

2.1 Output Probabilities of Shannon Co-Factors

The Shannon decompositions are very useful for describing the structure of Decision Diagram (DD) representations of Boolean functions. The DD form is often defined in terms of Shannon decompositions by considering subgraphs to be representative of Shannon co-factors of the diagrammed function [9]. It is convenient to describe conditional output probabilities by examining the output probabilities of co-factor functions as used in the Shannon expansion. The Shannon expansion theorem is well known and can be used to define co-factors. It is given in Equations 2, 3, and 4.

$$f(x_1, x_2, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_n) = \overline{x}_i f_0 + x_i f_1 \quad (2)$$

$$f_{\overline{x}_i} = f(x_1, x_2, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \quad (3)$$

$$f_{x_i} = f(x_1, x_2, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n) \quad (4)$$

Consider the co-factor, f_{x_i}, for a particular variable, x_i. The output probability is computed as the total number of times $f_{x_i} = 1$ divided by 2^{n-1} if f is a completely specified function of n variables. In terms of probability theory, this becomes the conditional probability value, $\wp\{f|x_i\}$. Using Baye's Rule the conditional probability may be expressed as given in Equation 5. Note that $\wp\{x_i\} \neq 0$ is always necessarily true when Equation 5 is used. This is due to the fact that the co-factors are always computed in terms of dependent x_i.

$$\wp\{f|x_i\} = \frac{\wp\{f \cdot x_i\}}{\wp\{x_i\}} \quad (5)$$

For fully specified functions with dependent variables that have equally likely probability distributions, $\wp\{x_i\} = \frac{1}{2}$, and Equation 5 simplifies to $\wp\{f|x_i\} = 2 \cdot \wp\{f \cdot x_i\}$. The use of circuit output probabilities in digital systems engineering tasks in the past has generally centered around the computation of the probabilities of single functions. In this paper we use conditional output probability expressions to show how the modified Haar transform spectral coefficients may be computed using a decision diagram. Unless otherwise noted, the conditioned quantity will always be dependent upon the function of interest. Therefore, the probability space for the conditional output probability, $\wp\{f|f_c\}$, has a size of 2^{n-m} where n is the number of dependent variables of f and m is the number of mutually dependent variables of f and f_c.

3 Computation of the Modified Haar Spectrum

This section will describe how output probabilities can be used to compute the modified Haar spectral coefficients directly. The idea is developed by making observations about the structure of the transformation matrix. Although actual vector-matrix products are not computed, this viewpoint is convenient in the development of the technique.

Each transformation matrix row consists of the integer elements -1, 1, and 0. -1 represents the Boolean 1 constant, 1 represents the Boolean 0 constant, and 0 indicates the absence of a Boolean constant. Each row represents some function, f_c, dependent upon n or fewer variables where n is the number of dependent variables of f, the function to be transformed.

3.1 Development of the Technique

Figure 1 contains the modified Haar transformation matrix for a function of $n = 3$ dependent variables. It is noted that higher ordered coefficients are computed from matrix row functions with a decreasing range space dimension. In fact, this decrease in the dimension of the range space corresponds directly to subsequent Shannon co-factors of the function to be transformed, f. This is in contrast to the results of similar observations for the Walsh and Reed-Muller transformation matrices where the respective f_c functions can be defined with total independence of the function to be transformed. One reason for this difference in row vector definition is that an individual spectral coefficient does not necessarily provide totally global information about the transformed function, rather it gives information regarding the correlation of the function with its various co-factors. Fortunately, like the previous approaches, each Haar coefficient can be computed as an algebraic relation of various probability values and hence the Haar spectrum is also directly linked with output probability calculations.

The output vector of the function to be transformed generally contains integers with -1 representing logic-1 and +1 representing logic-0. With this viewpoint, we can define the number of matches between a particular transformation matrix row vector as the number of times the row vector and function vector components are simultaneously equal to -1 or +1. Each of the constituent Boolean functions are given to the left of their respective transformation matrix row vectors. Note that the matrix contains a 0 value in addition to the 1 and -1 quantities which represent logic levels. Since some of the rows represent constituent functions that are co-factors, the output space is less than 2^3 in size and the presence of a 0 value acts as a place holder.

$$\begin{array}{r} f \\ x_1 \\ x_2 \cdot f_{\overline{x}_1} \\ x_2 \cdot f_{x_1} \\ x_3 \cdot f_{\overline{x}_1\overline{x}_2} \\ x_3 \cdot f_{\overline{x}_1 x_2} \\ x_3 \cdot f_{x_1\overline{x}_2} \\ x_3 \cdot f_{x_1 x_2} \end{array} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

Figure 1: Example of Modified Haar Transformation Matrix for $n = 3$

The presence of co-factors in the Haar constituent functions can be accounted for by using the relationship in Equation 5 to represent these quantities as output probabilities of the AND of the function to be transformed with its respective dependent literals. Note also that the maximum absolute value of a Haar spectral coefficient varies depending on the order of the coefficient. This is due to the reduction in the size of the range of the constituent functions containing co-factors.

In order to determine the total number of matching outputs between f and f_c, it is necessary to determine when both simultaneously evaluate to a logic-0 level as well as a logic-1 level. We denote the percentage of the total number of matches of logic-0 between some f and f_c as p_{m0} and likewise for the logic-1 levels, p_{m1}. With this viewpoint, f_c expressions can be constructed (shown to the left of the transformation matrix in Figure 1) that utilize co-factors of the function to be transformed to restrict the range space and to dictate where the relative location of the valid output of the f_c function occurs in the 2^n row vector components. Each of the co-factors is then ANDed with an appropriate co-factor to provide sign information (or alternatively, Boolean constant values) for the matrix row vector components.

The computation can now be viewed as finding the p_{m0} and p_{m1} values. This is true because each individual Haar spectral coefficient can be computed using the relationship in Equation 6.

Theorem 1 *The kth modified Haar spectral coefficient can be calculated as:*

$$H_k = 2^{n-i}[2(p_{m0} + p_{m1}) - 1] \tag{6}$$

Where n is the dimension of the range space of the function to be transformed, f, and i is the dimension of the range space of a particular Shannon co-factor of f. □

Proof: If N_m represents the number of times a scalar product value of +1 occurs in the computation of a particular modified Haar spectral coefficient (corresponding to 1×1 and -1×-1 products) and N_{mm} corresponds to the number of times a scalar product value of -1 occurs (corresponding to 1×-1 products), then the kth modified Haar spectral coefficient is given as:

$$H_k = N_m - N_{mm} \tag{7}$$

It is noted that the sum of N_m and N_{mm} must necessarily equal 2^{n-i} where i indicates the number of variables about which co-factors have been taken. Substituting this observation into Equation 7 yields:

$$H_k = 2N_m - 2^{n-i} \tag{8}$$

We define p_m to be the total percentage of times that a matching output between the f and f_c functions occur, therefore $p_m = 2^{n-i} \times N_m$. Furthermore, $p_m = p_{m0} + p_{m1}$. Substituting these definitions into Equation 8 yields the result:

$$H_k = 2^{n-i}[2(p_{m0} + p_{m1}) - 1] \tag{9}$$

□

The result of Theorem 1 reduces the computation of a single modified Haar spectral coefficient to that of finding matching percentages of identical similar outputs of f and f_c. This can be accomplished by applying the output probability computation algorithm to an OBDD representation of the functions representing the logic-0 matches and the logic-1 matches yielding p_{m0} and p_{m1} respectively. Using the result in Equation 5, the Shannon co-factor output probabilities can be computed by ANDing various cubes with the original function f and dividing the result by the output probability of the cube itself, which is a constant.

The following table contains symbols for each of the Haar spectral coefficients, H_i, values that indicate the size of the co-factor function range, i, and probability expressions that evaluate whether the function to be transformed and the constituent function simultaneously evaluate to logic-0 (denoted as p_{m0}), or evaluate to logic-1 (denoted as p_{m1}).

Table 1: Relationship of the Haar Spectrum and Output Probabilities

SYMBOL	i	$n-i$	p_{m1}	p_{m0}
H_0	0	3	$\wp\{f \cdot 0\}$	$\wp\{\overline{f} \cdot \overline{0}\}$
H_1	0	3	$\wp\{f \cdot x_1\}$	$\wp\{\overline{f} \cdot \overline{x}_1\}$
H_2	1	2	$\frac{\wp\{f \cdot \overline{x}_1 \cdot x_2\}}{\wp\{\overline{x}_1\}}$	$\frac{\wp\{\overline{f} \cdot \overline{x}_1 \cdot \overline{x}_2\}}{\wp\{\overline{x}_1\}}$
H_3	1	2	$\frac{\wp\{f \cdot x_1 \cdot x_2\}}{\wp\{x_1\}}$	$\frac{\wp\{\overline{f} \cdot x_1 \cdot \overline{x}_2\}}{\wp\{x_1\}}$
H_4	2	1	$\frac{\wp\{f \cdot \overline{x}_1 \cdot \overline{x}_2 \cdot x_3\}}{\wp\{\overline{x}_1 \cdot \overline{x}_2\}}$	$\frac{\wp\{\overline{f} \cdot \overline{x}_1 \cdot \overline{x}_2 \cdot \overline{x}_3\}}{\wp\{\overline{x}_1 \cdot \overline{x}_2\}}$
H_5	2	1	$\frac{\wp\{f \cdot \overline{x}_1 \cdot x_2 \cdot x_3\}}{\wp\{\overline{x}_1 \cdot x_2\}}$	$\frac{\wp\{\overline{f} \cdot \overline{x}_1 \cdot x_2 \cdot \overline{x}_3\}}{\wp\{\overline{x}_1 \cdot x_2\}}$
H_6	2	1	$\frac{\wp\{f \cdot x_1 \cdot \overline{x}_2 \cdot x_3\}}{\wp\{x_1 \cdot \overline{x}_2\}}$	$\frac{\wp\{\overline{f} \cdot x_1 \cdot \overline{x}_2 \cdot \overline{x}_3\}}{\wp\{x_1 \cdot \overline{x}_2\}}$
H_7	2	1	$\frac{\wp\{f \cdot x_1 \cdot x_2 \cdot x_3\}}{\wp\{x_1 \cdot x_2\}}$	$\frac{\wp\{\overline{f} \cdot x_1 \cdot x_2 \cdot \overline{x}_3\}}{\wp\{x_1 \cdot x_2\}}$

By observing that the probability expressions for a given H_k in Table 1 are statistically independent, the individual computations may be combined into a compact form. As an example, consider H_5.

Example 1 *The divisor for the p_{m0} and p_{m1} expressions, $\wp\{\overline{x}_1 x_2\}$ is a constant equal to $\frac{1}{2^i}$ and thus may be factored out resulting in Equation 9 being rewritten as:*

$$H_k = 2^{n-i}[2^{i+1}(p_{m0} + p_{m1}) - 1] \tag{10}$$

Since the Boolean expressions $f \cdot \overline{x}_1 \cdot x_2 \cdot x_3$ and $\overline{f} \cdot \overline{x}_1 \cdot x_2 \cdot \overline{x}_3$ are disjoint, the overall probability may be computed as the sum of the individual probabilities, or alternatively, as the probability of the inclusive-OR of the functions. This is true because it is easy to see that $\wp\{g + h\} = \wp\{g\} + \wp\{h\}$ for g and h that are covered by disjoint cube sets.

Combining the Boolean arguments and simplifying:

$$f \cdot \overline{x}_1 \cdot x_2 \cdot x_3 + \overline{f} \cdot \overline{x}_1 \cdot x_2 \cdot \overline{x}_3 = \overline{x}_1 x_2 (\overline{x_3 \oplus f}) \tag{11}$$

Therefore, we can rewrite Equation 10 as:

$$H_k = 2^{n-i}[2^{i+1}\wp\{\overline{x}_1 x_2 (\overline{x_3 \oplus f})\} - 1] \tag{12}$$

□

The manipulations used in Example 1 may be applied to all of the modified Haar spectrum coefficients. This leads to the interesting result that the modified Haar coefficients depend on the set of n Boolean functions, $\overline{x_i \oplus f}$, which which describe the equivalence of a particular dependent variable, x_i, and the function to be transformed, f. Higher ordered coefficients are based on disjoint partitions of the range space of these equivalence functions. The partitioning is accomplished by ANDing the equivalence functions with various cubes of other dependent variables of f. Table 2 contains the probability functions for a $n = 3$ variable transformation in terms of the equivalences. Using this table, each coefficient can be computed using Equation 13.

$$H_k = 2^{n-i}[2^{i+1}p_m - 1] \tag{13}$$

4 Implementation and Example

Since the output probabilities depend on the set of $n+1$ equivalence relationships given in the set A in Equation 14, the OBDD representation of each is formulated.

$$A = \{\overline{0 \oplus f}, \overline{x_1 \oplus f}, \overline{x_3 \oplus f}, \ldots, \overline{x_n \oplus f}\} \tag{14}$$

Analogous to the OBDD variable ordering problm, a specified variable order is also implicit for any given

Table 2: Relationship of the Haar Spectrum and Equivalence Functions

SYMBOL	i	$n-i$	p_m
H_0	0	3	$\wp\{\overline{f \oplus 0}\} = 1 - \wp\{f\}$
H_1	0	3	$\wp\{\overline{f \oplus x_1}\}$
H_2	1	2	$\wp\{\overline{x}_1 \cdot \overline{f \oplus x_2}\}$
H_3	1	2	$\wp\{x_1 \cdot \overline{f \oplus x_2}\}$
H_4	2	1	$\wp\{\overline{x}_1 \cdot \overline{x}_2 \cdot \overline{f \oplus x_3}\}$
H_5	2	1	$\wp\{\overline{x}_1 \cdot x_2 \cdot \overline{f \oplus x_3}\}$
H_6	2	1	$\wp\{x_1 \cdot \overline{x}_2 \cdot \overline{f \oplus x_3}\}$
H_7	2	1	$\wp\{x_1 \cdot x_2 \cdot \overline{f \oplus x_3}\}$

modified Haar spectrum. Higher ordered coefficients are computed by partitioning the relevant equivalence relation into 2^m OBDDs covering disjoint portions of the range and finding their output probabilities. The value $m-1$ is the number of previously used equivalence relations in the set A. The partitioning is quite simple and consists of formulating all possible cubes of the m literals used in lower ordered coefficients and ANDing them with the OBDD representing the equivalence relation.

4.1 Example Computation

A simple 3-variable function will be used to illustrate the computation approach. Consider the function given by the truth table in Figure 2. The corresponding modified Haar transform is computed by the definition and shown in Equation 15.

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \\ 1 \\ -1 \\ -1 \\ -1 \end{bmatrix} = \begin{bmatrix} -2 \\ 2 \\ 0 \\ 2 \\ 2 \\ 2 \\ 2 \\ 0 \end{bmatrix} \quad (15)$$

Each of these coefficients may be computed using the expressions derived in the previous subsection. Table 3 contains the results of the computations using the method proposed here.

x_1	x_2	x_3	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Figure 2: Truth Table of the Function for Example Computation

Table 3: Relationship of the Haar Spectrum and Equivalence Functions

SYMBOL	i	$n-i$	p_m	H_k
H_0	0	3	$\frac{3}{8}$	-2
H_1	0	3	$\frac{5}{8}$	2
H_2	1	2	$\frac{1}{4}$	0
H_3	1	2	$\frac{3}{8}$	2
H_4	2	1	$\frac{1}{4}$	2
H_5	2	1	$\frac{1}{4}$	2
H_6	2	1	$\frac{1}{4}$	2
H_7	2	1	$\frac{1}{8}$	0

When implemented, the probability calculations shown in column 4 of Table 3 are comprised of forming the OBDD representing the argument of the $\wp\{\}$ function and applying the probability assignment algorithm. The probability assignment algorithm has a complexity of $O(N)$ where N is the total number of vertices in the data structure. Therefore, individual spectral coefficients may be computed very quickly since many functions are represented by OBDDs with a total number of vertices polynomial in n.

4.2 Experimental Results

The computationally intense portion of the method described here involves the calculation of the probabilities. The OBDDs representing the fundamental equivalence relations given in the set A in Equa-

tion 14 will generally contain more internal vertices than the AND of those same functions with product terms. Therefore, we computed the probability values of the fundamental equivalence relations for the *ISCAS* benchmark circuits $c432$ and $c880$ for outputs $421gat$ and $878gat$ respectively. Tables 4 and 5 contain the results of these computations.

The algorithm used to compute the probabilities is a recursive, depth-first method that actually computes path probabilities rather than node probabilities. By computing path probabilities, the temporal complexity of the algorithm is $O(N)$, where N is the number of OBDD vertices. These computations were performed on a 100 MHz SPARCstation 20 with 160 MB of memory. All computations required 1 second or less of CPU time.

The relevance of this data is that the value of the modified Haar transform coefficients for this function may all be computed in 1 CPU second or less since the first-ordered coefficients are algebraically related to the given probabilities. The higher ordered coefficients are computed by ANDing a cube term with the OBDDs represented in the Table 4 and will thus generally yield smaller OBDDs allowing the corresponding probabilities to be computed with a negligible amount of increased CPU time overhead. Therefore, we have reduced the amount of time to compute a modified Haar transform coefficient from that required to form an inner product of vectors of length 2^{36} for the $c432$ example or 2^{45} for the $c880$ example to a $O(N)$ traversal of an OBDD.

Table 4: Output Probabilities for $\overline{f \oplus x_i}$ and First-Order Modified Haar Spectral Coefficient for Circuit $c432$, Output $421gat$

INPUT, x_i	$\wp\{\overline{f \oplus x_i}\}$	H_1	SIZE
4gat	0.642	19.6×10^9	3972
11gat	0.511	1.59×10^9	7427
21gat	0.507	977.4×10^6	5633
27gat	0.503	515.7×10^6	4226
17gat	0.484	-2.07×10^9	5503
14gat	0.461	-5.32×10^9	3973
8gat	0.426	-10.1×10^9	4483
1gat	0.378	-16.7×10^9	6921

Table 5: Output Probabilities for $\overline{f \oplus x_i}$ and First-Order Modified Haar Spectral Coefficient for Circuit $c880$, Output $878gat$

INPUT, x_i	$\wp\{\overline{f \oplus x_i}\}$	H_1	SIZE
59gat	0.500	41.9×10^9	5056
126gat	0.499	-377.1×10^6	3150
146gat	0.499	-56.0×10^9	3681
183gat	0.498	-91.8×10^9	3292
177gat	0.496	-220.1×10^9	3465
91gat	0.495	-322.6×10^9	5191
219gat	0.406	-6.57×10^{12}	3114
228gat	0.399	-7.05×10^{12}	3113
268gat	0.380	-8.43×10^{12}	5342
210gat	0.379	-8.48×10^{12}	6029
159gat	0.338	-11.3×10^{12}	3104

5 Conclusion

This paper has examined the relationship between the conditional output probabilities of Boolean functions and the modified Haar transform. A methodology was implemented for the computation of the modified Haar spectral coefficients and experimental results are given. One very interesting result is the dependence of the spectrum upon $n+1$ equivalence relationships between the function to be transformed and each of its' dependent variables and the constant 0.

In the future, we plan to investigate problems that can possibly be solved more efficiently by exploiting the information present in the Haar spectrum.

Acknowledgment

I would like to thank Roger Moore for generating the experimental results given in this paper.

References

[1] N. Ahmed and K. R. Rao. *Orthogonal Transforms for Digital Signal Processing.* Springer-Verlag, New York, New York, 1975.

[2] E. M. Clarke, K. L. McMillan, X. Zhao, M. Fujita, and J. Yang. Spectral Transformations for Large Boolean Functions with Applications to Technology Mapping. *Proceedings of ACM/IEEE Design Automation Conference*, pages 54–60, 1993.

[3] B. J. Falkowski and C.-H. Chang. Efficient Algorithms for Forward and Inverse Transformations between Haar Spectrum and Binary Decision Diagram. *Proceedings of the International Phoenix Conference on Computers and Communications*, pages 497–503, 1994.

[4] A. Ghosh, S. Devadas, K. Keutzer, and J. White. Estimation of Average Switching Activity in Combinational and Sequential Circuits. *Proceedings of the ACM/IEEE Design Automation Conference*, pages 253–259, June 1992.

[5] S. L. Hurst, D. M. Miller, and J. C. Muzio. *Spectral Techniques in Digital Logic*. Academic Press, Orlando, Florida, 1985.

[6] B. Krishnamurthy and I. G. Tollis. Improved Techniques for Estimating Signal Probabilities. *IEEE Transactions on Computers*, vol. 38, no. 7:1245–1251, July 1989.

[7] K. P. Parker and E. J. McCluskey. Probabilistic Treatment of General Combinational Networks. *IEEE Transactions on Computers*, vol. c-24:668–670, June 1975.

[8] M. Pedram. Power Minimization in IC Design: Principles and Applications. *ACM Transactions on Design Automation of Electronic Systems*, vol. 1, no. 1, 1996.

[9] T. Sasao and M. Fujita (editors). *Representations of Discrete Functions*. Kluwer Academic Publishers, Boston, Massachusetts, 1996.

[10] R. S. Stanković, T. Sasao, and C. Moraga. *Chap. 2 of Representations of Discrete Functions, T. Sasao and M. Fujita, eds.* Kluwer Academic Publishers, Norwell, Massachusetts, 1996.

[11] M. A. Thornton and V. S. S. Nair. Efficient Calculation of Spectral Coefficients and Their Applications. *IEEE Trans. on CAD/ICAS*, vol. 14, no. 11:1328–1341, November 1995.

[12] M. A. Thornton and V. S. S. Nair. BDD Based Spectral Approach for Reed-Muller Circuit Realisation. *IEE Proceedings, Computers and Digital Techniques*, vol. 143, no. 2:145–150, to appear.

[13] C. Y. Tsui, M. Pedram, and A. Despain. Efficient Estimation of Dynamic Power Dissipation Under a Real Delay Model. *Proceedings of the International Conference on Computer-Aided Design*, pages 224–228, June 1993.