Fixed Polarity Reed-Muller Expansions for 3-valued Logic

Mark MacIsaac
Department of Math. Stats.
Computer Science
St. Francis Xavier University
Antigonish, Nova Scotia, CANADA

Gerhard W. Dueck
Faculty of Computer Science
University of New Brunswick
Fredericton, N.B., CANADA

Abstract

Any n-variable symmetric function can be represented by a carrier vector of length $n+1$. For such functions, Suprun [5] has developed a method to find all fixed polarity Reed-Muller (FPRM) expansions from a simple triangle consisting of zeros and ones. The triangle is very compact, it only requires $(n+2)(n+1)/2$ bits, where n is the number of variables. Butler et al. [1] developed an algorithm that can construct the triangle and find the best polarity in $O(n^3)$ time. This is a significant improvement over the previous algorithm with complexity $O(n^7)$.

In this paper we extend the binary case to 3-valued logic. It is not surprising, that the complexity is greatly increased. The simple triangle becomes a complex multi-dimensional structure. Nevertheless we show that using this structure, significant savings in storage and processing time can be achieved.

1 Introduction

Symmetric function play an important role in circuit design. Many arithmetic function are symmetric. symmetric functions are more efficiently implemented in AND-EXOR networks, rather than AND-OR networks. AND-EXOR networks are more easily tested. Symmetric function are not easily minimized by Quine-McCluskey like algorithms [2]. They are therefore ideally suited for benchmark function, since they are difficult to minimize, yet they minimal representation is known. Binary symmetric function have been studied in depth [3, 4, 5, 1].

FPRM expansions are special class of AND-EXOR circuits. Suprun [5] has developed a unique structure from which all FPRM expansions can be obtained. His work has been expanded by Butler, Dueck, Shmerko, and Yanushkevich [1]. In this paper we review the binary case and propose some ways in which it can be extended to the ternary case. The complexity of the ternary case is significantly greater than for the binary one.

Position	Value	
000	$\bar{x}_1\bar{x}_1\bar{x}_3$	0
001	$\bar{x}_1\bar{x}_3x_3$	1
010	$\bar{x}_1x_2\bar{x}_3$	1
011	$\bar{x}_1x_2x_3$	0
100	$x_1\bar{x}_3\bar{x}_3$	1
101	$x_1\bar{x}_3x_3$	0
110	$x_1x_2\bar{x}_3$	0
111	$x_1x_2x_3$	1

Table 1: Table for Example 1

2 Symmetric Functions

A function $f(x_1, x_2, ..., x_n)$ is totally symmetric if and only if it is unchanged by any permutation of variables.

Example 1 *Consider a 3-variable binary function* $f(x_1, x_2, x_3)$ *with the truth vector* $T = [0,1,1,0,1,0,0,1]$. *From Table 1, it is seen that* $f(0,0,1) = f(0,1,0) = f(1,0,0)$ *and* $f(0,1,1) = f(1,0,1) = f(1,1,0)$. *Therefore, the function* $f = \bar{x}_1\bar{x}_2x_3 + \bar{x}_1x_2\bar{x}_3 + x_1\bar{x}_2\bar{x}_3 + x_1x_2x_3$ *is a symmetric function.*

For all symmetric functions, values are going to be repeated in the truth vector. In the previous example the value 1 for $f(0,0,1)$, $f(0,1,0)$ and $f(1,0,0)$ is represented 3 times in the truth vector, T. This repetition of values of a symmetric function can be taken advantage of by representing the function, f, as reduced, compact vector, called a *carrier vector*.

Definition 1 *A carrier vector is a* $n+1$ *size vector representing a totally symmetric function in binary.*

Example 2 *In the previous example the carrier vector* C, *would be* $[0,1,0,1]$.

3 Binary

3.1 Fixed Polarity Reed-Muller Expansion (FPRM)

The fixed polarity Reed-Muller expansion in polarity c for a function $f(x_1, x_2, ..., x_n)$ is:

$$\mathrm{RM}_c = g_0 \oplus g_1x_1^* \oplus g_2x_2^* \oplus ..... \oplus g_2x_n^* \oplus g_{n+1}x_1^*x_2^* \oplus ..... \oplus g_{2^n-1}x_1^*x_2^*...x_n^*$$

where x_i^* is either x_i or $\bar{x}_i$ everywhere. The term *fixed polarity* means every variable occurs in one polarity, x_i or $\bar{x}_i$. There are 2^n different polarities in which a function f can be represented. These different polarities depend on

Polarity	number of variables complemented	variable(s) complemented
0 (000)	0	—
1 (001)	1	x_1
2 (010)	1	x_2
3 (011)	2	x_1, x_2
4 (100)	1	x_3
5 (101)	2	x_1, x_3
6 (110)	2	x_2, x_3
7 (111)	3	x_1, x_2, x_3

Table 2: Different Polarities for 3-variable function

which variables are complemented. The question that arises, is which polarity has the fewest number of terms?

For a 3-variable function $f(x_1, x_2, x_3)$, the different polarities are listed, in table 2. They are represented in both there decimal and binary number.

Because a symmetric function is unchanged by a permutation of variables, it makes no difference which variables is RM_p appear complemented. Therefore the polarity RM_{001} will have the same number of terms as RM_{010}, so only one of these RM_p's have to be investigated. Therefore, instead of investigating 2^n polarities, only n+1 polarities have to be considered.

3.2 The Transeunt Triangle

Suprun [5] suggested the transeunt triangle as a means of comparing FPRM expansions. Butler *et al.* [1] developed an efficient algorithm for the synthesis of FPRM, based on the transeunt triangle. Here we present a brief review of its characteristics. The transeunt triangle is a structure that has the $n + 1$ carrier vector as its top row with each row having one less element then the next row. The bottom row has 1 element, therefore the transeunt triangle has n+1 rows with $\frac{(n+1)(n+2)}{2}$ elements. To form the transeunt triangle, start with the carrier vector $C = [c_0, c_1, ..., c_n]$. Each element of the next row, n, of size α, is formed using the elements of previous row, p (size $\alpha + 1$) by the following: $n_i = p_i \oplus p_{i+1}$ for $1 \leq i \leq \alpha$

The transeunt triangle contains the essential information for every FPRM for a given symmetric function, along with the carrier vector itself.

Example 3 *Consider a 4 variable totally symmetric function, f, with $C = [1, 0, 1, 1, 1]$. The generation of the triangle is shown in table 3.2. The transeunt triangle is for this function is:*

Row	Generated	Elements
1	carrier vector	1 0 1 1 1
2	$1 \oplus 0, 0 \oplus 1, 1 \oplus 1, 1 \oplus 1$	1 1 0 0
3	$1 \oplus 1, 1 \oplus 0, 0 \oplus 0$	0 1 0
4	$0 \oplus 1, 1 \oplus 0$	1 1
5	$1 \oplus 1$	0

Table 3: Generation of The Transeunt Triangle

$$\begin{array}{ccccccccc}
1 & & 0 & & 1 & & 1 & & 1 \\
& 1 & & 1 & & 0 & & 0 & \\
& & 0 & & 1 & & 0 & & \\
& & & 1 & & 1 & & & \\
& & & & 0 & & & &
\end{array}$$

The above transeunt triangle holds the information for all FPRM expansions for the given function f. As said earlier there are $n+1$ polarities for an n variable function. Therefore there are 5 polarities in this example. The carrier vector is the top row 10111. The 0000 polarity is given by the left side of the triangle:

$$\begin{array}{ccccc}
1 & & & & \\
& 1 & & & \\
& & 0 & & \\
& & & 1 & \\
& & & & 0
\end{array}$$

The next polarity, 0001, which also represents the 0010,0100,1000 polarities, is the following:

$$\begin{array}{ccccc}
& 0 & & & \\
1 & & 1 & & \\
& 0 & & 1 & \\
& & 1 & & 1 \\
& & & 0 &
\end{array}$$

Notice that 4 elements of the 0001 are shared with 4 elements of the 0000 polarity. The next polarity, 0011, is:

$$\begin{array}{ccccc}
& & 1 & & \\
& 1 & & 0 & \\
0 & & 1 & & 0 \\
& 1 & & 1 & \\
& & 0 & &
\end{array}$$

The polarity 1111 is the n+1 row on the right hand side of the triangle. Each polarity's first element is from the carrier vector which is not shared with any other polarity. As seen earlier the 0000 polarity is $[1, 1, 0, 1, 0]$. This represents

0000		0001		0011		0111		1111
1		0		1		1		1
	1		1		0		0	
		0		1		0		
			1		1			
				0				

Table 4: Starting Points for Polarities.

the function: $1 \oplus (x_1 \oplus x_2 \oplus x_3 \oplus x_4) \oplus (x_1x_2x_3 \oplus x_1x_2x_4 \oplus x_2x_3x_4)$. A nice property of this triangle is that you can start from any polarity and generate the triangle using the exor operation. Representing the FPRM of symmetric functions in this transeunt triangle form is a much more compact and easier way to produce the different FPRM of the function.

4 3-Valued Logic

4.1 3-Valued FPRM's

When dealing with 3-valued logic functions, the fixed polarity Reed Muller expansion is the same as the binary case except that x_i^* can be either x_i, $\widehat{x_i}$, or $\widehat{\widehat{x_i}}$, where $\widehat{x_i} = x_i \oplus 1$ and $\widehat{\widehat{x_i}} = x_i \oplus 2$ ($\oplus$ is addition modulus 3). For a given function, there are 3^n fixed polarities, which can be represented as $\frac{(n+1)(n+2)}{2}$ polarities when the function is symmetric. The carrier vector of a symmetric function is represented in triangular form. We refer to each value as its first assignment of values. For example, $f(1,1,2) = f(1,2,1) = f(2,1,1)$ is referred as 112, not 121 or 211. Below is an 3-variable example to illustrate the positions of values of the carrier vector in this form:

000		001		011		111
	002		012		112	
		022		122		
			222			

Therefore if $C = [1, 2, 0, 1, 2, 1, 1, 2, 0, 1]$, the triangle form would be:

1		2		1		1
	0		2		2	
		1		0		
			1			

The transeunt triangle took advantage of the sharing and symmetry which resulted in a compact representation. A 3-valued structure can also be formed, however it is more complex.

4.2 3-valued Logic Structure for FPRM Symmetric Functions

The 3 polarities with either all 0,1, or 2's (*i.e.* 0000,1111,2222) form a triangle in the same form as the carrier vector. Each having $\frac{(n+1)(n+2)}{2}$ elements. We refer to them as *triangle polarities.*

Given the elements for a polarity β we generate the next polarity by taking some elements from β and generate the new elements from these.

Definition 2 *A sub-triangle is defined as:*

$$\begin{array}{ccc} x_{i,j} & & x_{i,j+diff} \\ & x_{i+1,j} & \end{array}$$

where $x_{i,j}$ is the jth element in the ith row of β and where diff is the difference between the new polarity and previous triangle polarity (either 000,111,222, for a 3-variable function). For example, if the new polarity is 001, diff is 1, or the new polarity is 011 the diff is 2. When we reach a triangle polarity, we use the whole triangle β

Starting from a triangle polarity, there are two transform triangles needed to produce the next polarity.

$$A = \begin{array}{cc} 1 & 2 \\ 1 & \end{array}$$

and

$$B = \begin{array}{cc} 0 & 1 \\ 1 & \end{array}$$

The subtriangles from row_i and row_{i+1} of β form row_i the elements of next polarity. Y represents the number of subtriangles for each row_i and row_{i+1} of β. Using A and B, each new row_i of the next polarity, n, can be formed from β, using the following algorithm:

for $1 \leq i \leq \frac{y}{diff}$
  for $1 \leq j \leq$ diff
    new element=$(subtriangle^v_{1,1} * A_{1,1}) \oplus (subtriangle^v_{1,2}) * A_{1,2}) \oplus$
    $(subtriangle^v_{2,1} * A_{2,1})$

    new element=$(subtriangle^v_{1,1} * B_{1,1}) \oplus (subtriangle^v_{1,2}) * B_{1,2}) \oplus$
    $(subtriangle_{2,1} * B_{2,1})$

with $v \in (0, 1, ..., y)$

Example 4 *Consider a 3-variable with the polarity 000 as follows:*

$$\begin{array}{ccccccc} 1 & & 1 & & 2 & & 1 \\ & 0 & & 1 & & 2 & \\ & & 2 & & 1 & & \\ & & & 1 & & & \end{array}$$

Using transform triangle A and B and the top left-hand corner sub-triangle $\begin{smallmatrix}1 & & 1\\ & 0 & \end{smallmatrix}$ *of polarity 000.*

The new elements of polarity 001 can be formed the following way:

$newelement1 = (1*A_{1,1}) \oplus (1*A_{2,1}) \oplus (0*A_{2,1}) = (1*1) \oplus (1*2) \oplus (0*1) = 0$

$nextelement = (1*B_{1,1}) \oplus (1*B_{2,1}) \oplus (1*B_{2,1}) = (1*0) \oplus (1*1) \oplus (0*1) = 1$

The next subtriangle would be $\begin{smallmatrix}1 & & 2\\ & 1 & \end{smallmatrix}$.

If you rotate the 000 polarity 90 degrees so the values go into the paper, the polarity would look like:

$$\begin{array}{cccc} 1 & & & \\ & 0 & & \\ & & 2 & \\ & & & 1 \end{array}$$

with only the first element visible. If the polarity is then represented by the number or elements in each row, the new representation would be:

$$\begin{array}{cccc} 4 & & & \\ & 3 & & \\ & & 2 & \\ & & & 1 \end{array}$$

With the polarities formed from 000 to 111 (001 and 011), using the transform triangles, the structure would be:

$$\begin{array}{ccccccc} 4 & & 6 & & 6 & & 4 \\ & 3 & & 4 & & 3 & \\ & & 2 & & 2 & & \\ & & & 1 & & & \end{array}$$

Using the algorithm on the previous polarity elements [6,4,2], [6,3] is generated. Remember, these numbers are not values, they are the number of elements in each row. It is easier to show the structure this way, since it is a three-dimensional object. The polarities are then obtained in the same way as the transeunt triangle.

The 001 polarity would be

$$\begin{array}{ccccc} & 6 & & & \\ 3 & & 4 & & \\ & 2 & & 2 & \\ & & 1 & & \end{array}$$

The actual values of this polarity would be:

$$
\begin{array}{cccccccccccc}
0 & & 1 & & 0 & & 0 & & 0 & & 0 & (6) \\
 & & 1 & & 0 & & 0 & & 0 & & & (4) \\
 & & & & 2 & & 2 & & & & & (2) \\
 & 0 & & & & 1 & & & & 2 & (3) & \\
 & & & 2 & & & & 1 & & & (2) & \\
 & & & & & 1 & & & & & (1) &
\end{array}
$$

The structure has the form

$$
\begin{array}{ccccccc}
4 & & & & & & 4 \\
 & 3 & & & & 3 & \\
 & & 2 & & 2 & & \\
 & & & 1 & & & \\
 & & & 2 & & & \\
 & & & 3 & & & \\
 & & & 4 & & &
\end{array}
$$

where each vector $[4, 3, 2, 1]$ represents a triangle polarity and the elements which make up the other polarities are between them. Between 000 and 111, would be 001 and 011, between 111 and 222 would be 112 and 122, and between 222 and 000 would be 220 and 200. From one polarity to the next only one bit in the name can change.

Each tri-dent would have the same number of elements. Therefore each tri-dent would have a top row of $[4, 6, 6, 4]$. The top element of each of these 10 rows ($[4, 6, 6, 4, 6, 6, 4, 6, 6]$ is a part of the carrier vector, this is very similar to the transeunt triangle.

This paper deals with 3-valued Logic. We further restrict the discussion to polarities which have only 2 different values (*i.e.*, 011 or 122, or 220). The polarities with all 3 values are still being investigated.

5 Conclusion

In this paper we reviewed the FPRM for binary functions and outlined some of the ways in which this can be extended to ternary logic. The complexity of the ternary case is significantly greater than the binary one. More work is needed to fully describe the translations required from one triangle to the next one. The structure for the ternary "triangle" is not clearly defined. A data structure for its implementation has to be defined. However the advantages are also significant, since the alternative is to store all coefficients and transform them via matrices.

References

[1] J. T. Butler, G. W. Dueck, V. P. Shmerko, and S. N. Yanushkevich. An efficient algorithm for the synthesis of fixed polarity Reed-Muller expansions

of symmetric functions. *submitted to IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.*

[2] D. L. Dietmeyer. Generating minimal covers of symmetric functions. *IEEE Trans. on Computer Aided Design of Integrated Circuits and Systems*, 12(5):710–713, 1993.

[3] R. Drechsler and B. Becker. Sym*pathy*: Fast exact minimization of fixed polarity Reed-Muller expansion for symmetric functions. *IEEE Trans. on Computer Aided Design of Integrated Circuits and Systems*, 16(1):1–5, 1997.

[4] C. Scholl, D. Moller, P. Molitor, and R. Drechsler. BDD minimization using symmetries. *IEEE Trans. on Computer Aided Design of Integrated Circuits and Systems*, 18(2):81–99, 1999.

[5] V. P. Suprun. Fixed polarity reed-muller expressions of symmetric boolean functions. In *Proc. IFIP WG 10.5 Workshop on Application of the Reed-Muller Expansions in Circuit Design, Japan*, pages 246–249, 1995.