Antisymmetries in the Representation of Boolean and Multi-Valued Functions

submitted to ISMVL2005

J. E. Rice
Dept. of Math & Computer Science
University of Lethbridge
Lethbridge, Alberta, CANADA
j.rice@uleth.ca

J. C. Muzio
Department of Computer Science
University of Victoria
Victoria, British Columbia, CANADA
jmuzio@cs.uvic.ca

Abstract

Detection methods for a class of symmetries called anti-symmetries *are presented. Additionally, logic synthesis applications of these symmetries are discussed. Both topics are extended from binary to multi-valued logic.*

1. Introduction

Partial symmetries exist in many functions, particularly those used in practical applications. Both total and partial symmetry properties are commonly used in synthesis of digital circuits [4, 6].

There is various literature on partial and total symmetries in binary logic such as [4, 9, 3]. However, most of the documented symmetry properties depend on the identification of two identical subfunctions within a Boolean function. In this paper we examine partial symmetries that occur when one of the subfunctions is not identical to the other, but is instead the inverse of the other. We refer to these as *anti-symmetries*, certain types of which were introduced by Tsai and Marek-Sadowska in [16] as *skew*-symmetries and by Falkowski and Kannurao in [5] as *complement single variable* symmetries.

Due to the increasing popularity of multiple-valued logic, it seems logical to extend these concepts from the binary to the multiple-valued domain. Symmetries in multiple-valued logic have been discussed in various places in the literature such as [10, 1, 2, 11, 15]. [1] and [2] discuss only the totally symmetric case, while the remainder of these references limit their discussions to permutations of the function's variables. or of the variable values. We extend the idea of partial (anti)symmetries of degree two to multiple-valued logic.

2. Preliminaries

In this section essential notation and background is presented. We assume throughout this paper that all functions are completely specified.

2.1. Symmetries

A function f is said to be symmetric with respect to a set $\lambda \subseteq \mathcal{V}_n$ if f remains unchanged for all permutations of the variables in λ. If $\lambda = \mathcal{V}_n$ then we say that the function is totally symmetric, otherwise we say that it is partially symmetric over the variables in λ [8].

A symmetry of degree two is a partial symmetry in which the two subfunctions that are identical are independent of two of the function's variables. Antisymmetries are based on Hurst *et. al's.* [4] definitions of equivalence, nonequivalence and single-variable symmetries, which are all symmetries of degree two.

A Boolean function f is said to possess an *equivalence* symmetry if there exist two variables $\{x_i, x_j\} \subseteq \mathcal{V}_n$ such that

$$f(x_{n-1}, \dots, 0, \dots, 0, \dots, x_0) = f(x_{n-1}, \dots, 1, \dots, 1, \dots, x_0)$$

Subfunctions such as these that are based on fixing two variables are used throughout this paper, and so here we define a shortened notation. Without loss of generality we label the variables to be fixed as $\{x_{n-1}, x_{n-2}\}$ allowing us to simplify the definitions and use the following notation:

$$f_u = (u_{n-1}, u_{n-2}, x_{n-3}, \dots, x_0)$$

where

$$u = \sum_{i=0}^{n-m-1} u_{m+i} \cdot p^i$$

In this definition n is the number of variables for the p-valued function and $n-m$ is the number of variables whose values are fixed, which in this paper is 2.

Using this shortened notation the equivalence symmetry can then be defined as $f_0 = f_3$. and is written $E\{x_{n-1}, x_{n-2}\}$. *Non-equivalence* symmetries are written $N\{x_{n-1}, x_{n-2}\}$ and are defined as $f_1 = f_2$, and a third type of symmetry called *single-variable* symmetries are written $S\{x_{n-2}|x_{n-1}\}$ or $S\{x_{n-2}|\overline{x}_{n-1}\}$ and are defined as $f_2 = f_3$ or $f_0 = f_1$.

2.2. Spectral Coefficients

In [10] Miller extends a number of spectral tests presented by Hurst to multiple-valued logic. Here we provide an overview of this work; however, the reader is directed to the original paper for details.

As an alternative to the p^n entries necessary to define a function's truth table we may also specify p^n coefficients that define the function, these coefficients being the spectral coefficients. In our discussion of the spectral coefficients we will adopt Miller's strategy of a matrix notation.

The spectral coefficient vector for a p-valued function f is defined as follows: let $f(v)$ denote $f(x_{n-1}, ..., x_0)$, where $x_i = v_i$, $0 \leq i \leq n-1$ and $v = \sum_{i=0}^{n-1} v_i p^i$. Then

$$s_w = \sum_{v=0}^{p^n-1} t_w(v) y(v)$$

where $y(v) = a^{f(v)}$, $a = e^{-j\frac{2\pi}{p}}$ and $j = \sqrt{-1}$. We write this as

$$S = T^n \cdot Y \tag{1}$$

where Y is the vector whose v^{th} element is $y(v)$, S is the vector whose w^{th} element is s_w, and T^n is a matrix whose element in the w^{th} row and v^{th} column is $t_w(v)$. If $p = 2$ then the rows of T^n are the Rademacher-Walsh functions. If $p > 2$ then they are the Chrestenson functions.

Based on the subfunctions described in Section 2.1 we denote Y_u as the output vector for f_u. We can order Y such that

$$Y = \begin{matrix} Y_0 \\ Y_1 \\ \vdots \\ Y_\beta \end{matrix}$$

where $\beta = 2^{n-m} - 1$. We can partition S similarly into $S^0, S^1, \ldots, S^\beta$ where S^u consists of s_w for which $w_i = u_i, m \leq i \leq n-1$ and each s^u is then a subvector with p^m elements.

An alternative spectral coefficient vector may be computed using the following definition:

$$S_u = T^{n-2} \cdot Y_u$$

where Y_u is the output vector resulting from subfunction f_u. The relationship between S_u and S^u is

$$[S_0 S_1 \cdots S_\beta] = \frac{1}{p^{n-m}} \left[S^0 S^1 \cdots S^\beta\right] \overline{T^{n-m}} \tag{2}$$

where $\overline{T^{n-m}}$ is the complex conjugate of T^{n-m}. These partitions will be used in Section 6 in demonstrating conditions and tests for the existence of antisymmetries.

3. Antisymmetries

The equivalence, nonequivalence and single-variable symmetries are specific types of partial symmetries of degree two for binary logic. In [13] we have extended this to the notion of an *antisymmetry*. Antisymmetries exist when a Boolean function f possesses two subfunctions that are the exact inverse of each other. For instance, a Boolean function f is said to possess an *anti-equivalence* symmetry $\overline{E}\{x_{n-1}, x_{n-2}\}$ if

$$f(0, 0, x_{n-3}, \ldots, x_0) = \overline{f}(1, 1, x_{n-3}, \ldots, x_0)$$

Table 1 summarizes the various types of antisymmetries in Boolean functions and their definitions.

Antisymmetry	Definition
$\overline{E}\{x_{n-1}, x_{n-2}\}$	$f(0, 0, x_{n-3}, \ldots, x_0) = \overline{f}(1, 1, x_{n-3}, \ldots, x_0)$
$\overline{N}\{x_{n-1}, x_{n-2}\}$	$f(0, 1, x_{n-3}, \ldots, x_0) = \overline{f}(1, 0, x_{n-3}, \ldots, x_0)$
$\overline{S}\{x_{n-2}|x_{n-1}\}$	$f(1, 0, x_{n-3}, \ldots, x_0) = \overline{f}(1, 1, x_{n-3}, \ldots, x_0)$
$\overline{S}\{x_{n-2}|\overline{x}_{n-1}\}$	$f(0, 0, x_{n-3}, \ldots, x_0) = \overline{f}(0, 1, x_{n-3}, \ldots, x_0)$

Table 1. Definitions and notation for the antisymmetries.

4. Two-Variable Symmetries in the Multiple-Valued Case

The notion of a two-variable symmetry can clearly be extended to the multi-valued case. Equivalence, nonequivalence and single-variable symmetries may also be present in p-valued functions. However, simply denoting the type of symmetry and variables involved is not sufficient, as there are p possible values that the variables may take. We use the following notation: $f_i = f_j$ is denoted

$$P_{i,j}\{x_{n-1}, x_{n-2}\}$$

where $i \neq j$ and $i, j \in \{0, \ldots, p^2 - 1\}$. i and j indicate the values assumed by x_{n-1} and x_{n-2} respectively. Again, we point out that this does not limit the symmetric variables to x_{n-1} and x_{n-2}; however, these variables were chosen for the sake of simplicity and consistency in the notation. It would certainly be possible to categorize these symmetries into types matching those defined for $p = 2$ as equivalence, non-equivalence and single-variable symmetries; however, this has not been deemed necessary for this work.

For example, if $p = 3$ and $n = 3$ then a possible two-variable symmetry can be expressed as $f_2 = f_4$, which may then be expanded to $f(0, 2, x_0) = f(1, 1, x_0)$. An example function with this partial symmetry is shown in Figure 1.

$$\begin{array}{cc|ccc} & & & x_2 & \\ & & 0 & 1 & 2 \\ \hline & 0 & - & - & - \\ x_1 & 1 & - & a & - \\ & 2 & a & - & - \end{array} \qquad \begin{array}{cc|ccc} & & & x_2 & \\ & & 0 & 1 & 2 \\ \hline & 0 & - & - & - \\ x_1 & 1 & - & b & - \\ & 2 & b & - & - \end{array}$$

$x_0 = 0$ $\qquad$ $x_2 = 1$

$$\begin{array}{cc|ccc} & & & x_2 & \\ & & 0 & 1 & 2 \\ \hline & 0 & - & - & - \\ x_1 & 1 & - & c & - \\ & 2 & c & - & - \end{array}$$

$x_0 = 2$

Figure 1. A function with a partial symmetry $P_{2,4}\{x_2, x_1\}$.

$$\begin{array}{cc|ccc} & & & x_1 & \\ & & 0 & 1 & 2 \\ \hline & 0 & a^{\uparrow 2} & - & - \\ x_0 & 1 & - & - & - \\ & 2 & - & a & - \end{array} \qquad \begin{array}{cc|ccc} & & & x_1 & \\ & & 0 & 1 & 2 \\ \hline & 0 & b^{\uparrow 2} & - & - \\ x_0 & 1 & - & - & - \\ & 2 & - & b & - \end{array}$$

$x_2 = 0$ $\qquad$ $x_2 = 1$

$$\begin{array}{cc|ccc} & & & x_1 & \\ & & 0 & 1 & 2 \\ \hline & 0 & c^{\uparrow 2} & - & - \\ x_0 & 1 & - & - & - \\ & 2 & - & c & - \end{array}$$

$x_2 = 2$

Figure 2. An example antisymmetry where $f(x_2, 1, 2) = f^{\uparrow 2}(x_2, 0, 0)$ for $n = 3$ and $p = 3$.

5. Antisymmetries in the Multiple-Valued Case

In order to extend these partial symmetries to the p-valued equivalent of our Boolean antisymmetries, the multiple-valued equivalent of the inverse, or negation operator must be used. Negation is often extended to multiple-valued logic as $\overline{x} = p - 1 - x$ where x is some p-valued variable, $x \in \{0, \ldots, p-1\}$. If a *balanced, symmetric* encoding of the function outputs is used then this may be redefined as $\overline{x} = -x, x \in \{-\lfloor \frac{p}{2} \rfloor, \ldots, \lfloor \frac{p}{2} \rfloor\}$.

However, we may also want to encompass other modifications of the literals, so we will also consider the following: the cyclic negation of a p-valued variable $x \in \{0, \ldots, p-1\}$ (or a function f) by k is denoted $x^{\uparrow k}$ and is defined as

$$x^{\uparrow k} = (x + k) \bmod p$$

It follows then, that in multiple-valued logic a number of possible antisymmetries may be defined. Again, some defining notation is required.

- A function f possesses an antisymmetry $P_{i,j}^{\uparrow k}\{x_{n-1}, x_{n-2}\}$ if $f_i = f_j^{\uparrow k}, k \in \{1, ..., p-1\}$.
- Alternatively, a function f may possess an antisymmetry $\overline{P}_{i,j}\{x_{n-1}, x_{n-2}\}$ if $f_i = \overline{f}_j$ using either of $\overline{x} = -x, x \in \{-\lfloor \frac{p}{2} \rfloor, \ldots, \lfloor \frac{p}{2} \rfloor\}$ or $\overline{x} = p - 1 - x, x \in \{0, \ldots, p-1\}$.

For example, Figure 2 illustrates a $p = 3$, $n = 3$ function with the antisymmetry $P_{5,0}^{\uparrow 2}\{x_1, x_0\}$.

A further comment on notation and labeling is pertinent. In the $p = 2$ case, $v = (v^{\uparrow 1})^{\uparrow 1}$; or alternatively, $v = \overline{\overline{v}}$. However, for most combinations of $p > 2$ and $k \neq 0$, $v \neq (v^{\uparrow 1})^{\uparrow 1}$. Thus it may be necessary to indicate in the notation for the p-valued cyclic antisymmetry which subfunction is the operand for the cyclic negation operator. This will be considered in future work.

6. Conditions and Tests for the Antisymmetries in Boolean Functions

In this section we present both conditions and tests for the antisymmetries of Boolean functions, based on the function's spectral coefficients. Until recently, the spectral coefficients were considered too expensive to compute and so this technique was not popular. However, Thornton *et. al.* have presented methods for efficient calculation of the spectral coefficients based on BDDs [14], thus making spectral techniques feasible for many practical functions.

Miller[10] shows that $f_i = f_j$ has the condition that $S_i = S_j$ and thus a test for the partial symmetries can be derived to be

$$\left[S^0 S^1 \cdots S^{\beta}\right] (\overline{T_i^{n-m}} - \overline{T_j^{n-m}}) = 0]$$

where $\overline{T_i^{n-m}}$ is the i^{th} column of $\overline{T^{n-m}}$ and $\overline{T_j^{n-m}}$ is the j^{th} column of $\overline{T^{n-m}}$.

Since $n - m = 2$ and the partition of $S = \left[S^0 S^1 \cdots S^{\beta}\right]$ can be described as follows:

> S^0 consists of all spectral coefficients s_w involving neither variable at u_{n-1} or u_{n-2};
>
> S^1 consists of all spectral coefficients s_w involving only the variable fixed at u_{n-2};
>
> S^2 consists of all spectral coefficients s_w involving only the variable fixed at u_{n-1};
>
> S^3 consists of all spectral coefficients s_w involving both variables fixed at u_{n-1} and u_{n-2},

we have

$$\left[S^0 S^1 S^2 S^3\right] (\overline{T_i^2} - \overline{T_j^2}) = 0]$$

For the antisymmetries this has to be modified, as we show below. For the antisymmetries the condition is $S_i = -S_j$ or equivalently $S_i + S_j = 0$, as negating a Boolean function also negates its spectral vector [4]. The test is therefore

$$\left[S^0 S^1 S^2 S^3\right] (\overline{T_i^2} + \overline{T_j^2}) = 0] \qquad (3)$$

PROOF:

From Equation 2 we have

$$\begin{aligned} S_i &= \tfrac{1}{p^{n-m}} \left[S^0 S^1 \cdots S^\beta\right] \overline{T_i^{n-m}} \\ S_j &= \tfrac{1}{p^{n-m}} \left[S^0 S^1 \cdots S^\beta\right] \overline{T_j^{n-m}}. \end{aligned}$$

To detect $S_i = S_j$ we look for $S_i - S_j = 0]$ and so we have the test of

$$\frac{1}{p^{n-m}} \left[S^0 S^1 \cdots S^\beta\right] (\overline{T_i^{n-m}} - \overline{T_j^{n-m}}) = 0]\,.$$

and the constant $\frac{1}{p^{n-m}}$ is dropped since the desired result is the 0 vector. This is a slight modification of the general test for (non anti-) symmetries presented in [10].

Since for the antisymmetries we wish to detect $S_i = -S_j$ we therefore test for $S_i + S_j = 0]$ and so

$$S_i + S_j = \frac{1}{p^{n-m}} \left[S^0 S^1 \cdots S^\beta\right] (\overline{T_i^{n-m}} + \overline{T_j^{n-m}}) = 0]\,.$$

■

7. Conditions and Tests for the Antisymmetries in Multiple-Valued Functions

The extension of these techniques to the multiple-valued case is non-trivial. We first examine a number of examples to illustrate the problem.

7.1. Antisymmetries Using Negation Based on Balanced Symmetric Encoding

The simplest case to examine is that of a function for which balanced, symmetric encoding of the outputs is used. In this case, we can illustrate with a small example that the use of individual columns of the transform matrix may still be utilized in detecting antisymmetries. Consider a function for which $p = 3$ and $n = 2$. Then we have

$$Y = \begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \end{bmatrix}, \quad S_i = T^{n-1} Y_i \quad \text{and} \quad S = \begin{bmatrix} S_0 \\ S_1 \\ S_2 \end{bmatrix}.$$

From (2) we have

$$\left[\begin{array}{ccc} S_0 & S_1 & S_2 \end{array}\right] = \frac{1}{3} \left[\begin{array}{ccc} S^0 & S^1 & S^2 \end{array}\right] \overline{T}^1.$$

where

$$\overline{T}^1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & a & a^2 \\ 1 & a^2 & a \end{bmatrix}.$$

Let us further define our function such that for some $i, j, i \neq j$, $Y_i = -Y_j$. Then $S_i + S_j = 0]$ which can be written as

$$\left[\begin{array}{ccc} S^0 & S^1 & S^2 \end{array}\right] \left[\overline{T}_i^1 + \overline{T}_j^1\right] = 0]\,.$$

This gives conditions for the antisymmetry on S^0, S^1, and S^2. For example, if $Y_0 = -Y_1$ then the condition is

$$\begin{aligned} \left[\begin{array}{ccc} S^0 & S^1 & S^2 \end{array}\right] \left[\overline{T}_0^1 + \overline{T}_1^1\right] &= 0] \\ \left[\begin{array}{ccc} S^0 & S^1 & S^2 \end{array}\right] \begin{bmatrix} 2 \\ 1+a \\ 1+a^2 \end{bmatrix} &= 0] \\ 2S^0 + (1+a)S^1 + (1+a^2)S^2 &= 0]\,. \end{aligned}$$

This then provides conditions for an antisymmetry of degree 1 where the antisymmetry is $f(0, x_0) = f(1, x_0)$. The other possible antisymmetries and their conditions would be

$$\begin{aligned} Y_0 = -Y_2 \quad &\leftrightarrow \quad 2S^0 + (1+a^2)S^1 + (1+a)S^2 = 0] \\ &\text{and} \\ Y_1 = -Y_2 \quad &\leftrightarrow \quad 2S^0 + (a+a^2)(S^1 + S^2) = 0]\,. \end{aligned}$$

We can then extend this to antisymmetries of degree 2. Let $n = 3$ and then from (2) we have

$$[S_0 \ldots S_8] = \frac{1}{9} \left[S^0 \ldots S_8\right] \overline{T}^2$$

where

$$\overline{T}^2 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & a & a^2 & 1 & a & a^2 & 1 & a & a^2 \\ 1 & a^2 & a & 1 & a^2 & a & 1 & a^2 & a \\ 1 & 1 & 1 & a & a & a & a^2 & a^2 & a^2 \\ 1 & a & a^2 & a & a^2 & 1 & a^2 & 1 & a \\ 1 & a^2 & a & a & 1 & a^2 & a^2 & a & 1 \\ 1 & 1 & 1 & a^2 & a^2 & a^2 & a & a & a \\ 1 & a & a^2 & a^2 & 1 & a & a & a^2 & 1 \\ 1 & a^2 & a & a^2 & a & 1 & a & 1 & a^2 \end{bmatrix} \qquad (4)$$

Again, the antisymmetry can be expressed as $Y_i + Y_j = 0]$ and so the conditions can be given by

$$\left[S^0 \cdots S^8\right] \left[\overline{T}_i^2 + \overline{T}_j^2\right] = 0]\,.$$

For example, if the antisymmetry is characterized by $Y_2 + Y_7 = 0]$ then

$$\begin{aligned} S_2 + S_7 &= 0] \\ \left[S^0 \cdots S^8\right] \left[\overline{T}_2^2 + \overline{T}_7^2\right] &= 0] \end{aligned}$$

and

$$(\overline{T}_2^2 + \overline{T}_7^2)' = (2a + a^2 a + a^2 1 + a^2 1 + a^2 2a 1 + a 2a^2 a + 1)'.$$

It is worth noting that Moraga [12] determined that if balanced, symmetric encoding of inputs and outputs is used, and if p is odd, $S'(w) = \overline{S(-w)}$ where $S(w)$ is the spectrum of the unaltered function and $S'(w)$ is the spectrum of the negation of the function. We are still investigating this distinction in our work.

7.2. Antisymmetries Using Cyclic Negation

The problem is much more difficult when cyclic negation is used. A similar example illustrates this. Let us define a function for which $p = 3, n = 2$, and $f(i) = f^{\uparrow 1}(i + 3)$ for $i \in \{3, 4, 5\}$. We will denote the outputs of the function as α_i. Then we have

$$\begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \\ s_8 \end{bmatrix} = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_3^{\uparrow 1} \\ \alpha_4^{\uparrow 1} \\ \alpha_5^{\uparrow 1} \end{bmatrix} \cdot \quad T^2$$

where T^2 is as defined above in (4). Then we have, for instance,

$$\begin{aligned} s_0 &= \alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_5^{\uparrow 1} + \alpha_6^{\uparrow 1} + \alpha_7^{\uparrow 1} \\ s_3 &= \alpha_0 + \alpha_1 + \alpha_2 + a\alpha_3 + a\alpha_4 + a\alpha_5 + a^2\alpha_5^{\uparrow 1} + a^2\alpha_6^{\uparrow 1} + a^2\alpha_7^{\uparrow 1} \\ s_6 &= \alpha_0 + \alpha_1 + \alpha_2 + a^2\alpha_3 + a^2\alpha_4 + a^2\alpha_5 + a\alpha_5^{\uparrow 1} + a\alpha_6^{\uparrow 1} + a\alpha_7^{\uparrow 1} \end{aligned}$$

and so on. $s_3 - s_6$ results in

$$\begin{aligned} &a\alpha_3 + a\alpha_4 + a\alpha_5 - a^2\alpha_3 - a^2\alpha_4 - a^2\alpha_5 \\ &+ a^2\alpha_5^{\uparrow 1} + a^2\alpha_6^{\uparrow 1} + a^2\alpha_7^{\uparrow 1} - a\alpha_5^{\uparrow 1} - a\alpha_6^{\uparrow 1} - a\alpha_7^{\uparrow 1} \end{aligned}$$

but any simplification is dependent on how the values are affected by the cyclic negation. Work is ongoing in this area.

8. Applications

There are, as indicated in the Introduction, many areas in which symmetries have played a part in improving upon some aspect of the logic synthesis process.

8.1. Decision Diagrams

In [11] a type of symmetries termed α-*symmetries* are introduced. A function $f(X)$ is α-symmetric in the variable pair (x_i, x_j) if there is no change when $x_i^{\uparrow \alpha}$ is substituted for x_j and $x_j^{\uparrow p-\alpha}$ for x_i, or

$$f(x_i, x_j, x_{n-3}, \ldots, x_0) = f(x_i^{\uparrow \alpha}, x_j^{\uparrow p-\alpha}, x_{n-3}, \ldots, x_0).$$

Clearly these are related to what we have identified as partial symmetries $P_{i,j}\{x_{n-1}, x_{n-2}\}$. The authors use α-symmetries in a technique to create symmetric-variable nodes in reduced, ordered MDDs (ROMDDs). The effect this has on the ROMDD is to reduce its *depth*, something generally not otherwise possible with traditional decision diagram manipulation and reduction techniques. We propose the addition of the notion of antisymmetries to this technique, to further leverage the (anti)symmetries inherent in many functions.

w		1			0			1		
y		-1	0	1	-1	0	1	-1	0	1
x	-1	-1	-1	-1	-1	0	1	-1	0	1
	0	0	0	0	0	0	1	0	0	1
	1	1	1	1	1	1	1	1	1	1

v=-1

w		-1			0			1		
y		-1	0	1	-1	0	1	-1	0	1
x	-1	1	0	-1	1	0	-1	1	0	-1
	0	0	0	-1	0	0	-1	1	0	-1
	1	-1	-1	-1	-1	-1	-1	1	0	-1

v=0

w		-1			0			1		
y		-1	0	1	-1	0	1	-1	0	1
x	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
	0	-1	-1	-1	-1	-1	-1	-1	-1	-1
	1	-1	-1	-1	-1	-1	-1	-1	-1	-1

v=1

Table 2. An example function where $p = 3$ and $n = 4$.

8.2. Logic Synthesis

The use of antisymmetries in logic synthesis can be best illustrated through the use of an example. Table 2 gives a function where $p = 3$ and $n = 4$. Clearly there are a number of symmetries and a number of antisymmetries, including $P_{1,2}\{v, w\}$, $\overline{P}_{2,4}\{v, w\}$ and $P_{3,4}\{v, w\}$. One possible realization making use of both the symmetries and antisymmetries is shown in Figure 3. The example uses min, max, inverters and literal gates. The literal gates are indicated with superscripts next to the variables. A max of min solution determined by the authors is

$$\begin{gathered} v^{-1}x^1 + v^0x^{-1}y^{-1} + v^0w^1y^{-1} + v^{-1}w^{0,1}y^1 \\ +0(v^0x^{-1,0}y^{-,1,0} + v^{-1,0}w^1y^0 + v^{-1}x^0 + v^{-1}w^0y^0) \end{gathered}$$

which has 8 terms and can be implemented with a fan-in cost of 33. In comparison, the sample realization in Figure 3 has a fan-in cost of 20. It additionally requires many fewer literal gates, although it makes use of inverters while the max of min solution does not. Note that in order to determine these implementations both the inputs and the outputs used balanced ternary encoding. For the sake of identifying antisymmetries (and to simplify the notation for the subfunctions) we assume that only the outputs are generally encoded in this manner, and that the input variables remain as values in $\{0, \ldots, p^n - 1\}$.

9. Conclusions

We have presented symmetries of degree two that are based on identification of subfunctions within a function that are the exact inverse of each other. Previous work identifies these antisymmetries in Boolean functions, while this paper extends the idea to the multiple-valued case. We find, however, that similarly extending the spectral tests and conditions is a difficult problem. Although computation of the spectrum for the Boolean case has been considerably improved by decision diagram techniques, it is not clear if

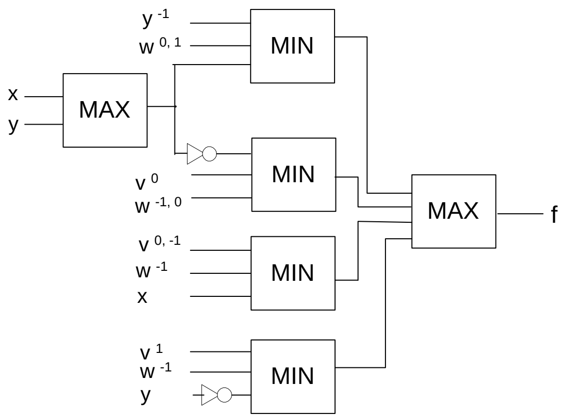

Figure 3. A possible realization using partial (anti)symmetries for the $n = 4$ $p = 3$ function specified in Table 2.

these techniques carry over into the multiple-valued case. Thus the determination of spectral conditions and tests for multi-valued antisymmetries may only be of use in analysis, rather than in practical applications.

We introduce two potential applications for multiple-valued antisymmetries; one in the ever-popular area of decision diagram reduction and one in the more traditional area of logic synthesis.

In addition to solving the problems yet unsolved in this paper, there are many other directions in which this work may evolve. These include extensions to symmetries of degree n, $n > 2$, and to incompletely specified functions. Additionally, it may be of interest to examine whether a symmetry of one or more particular cycle(s) are of greater or lesser use than other cycles in various areas of logic synthesis.

References

[1] J. T. Butler, D. S. Herscovici, T. Sasao, and R. J. Barton III. Average and Worst Case Number of Nodes in Decision Diagrams of Symmetric Multiple-Valued Functions. *IEEE Trans. on Comp.*, pages 491–494, 1997.

[2] J. T. Butler and T. Sasao. On the Properties of Multiple-Valued Functions That Are Symmetric in Both Variable Values and Labels. In *Proceedings of the 28th International Symposium on Multiple-Valued Logic (ISMVL)*, pages 83–88, 1998.

[3] L. Heinrich-Litan and P. Molitor. Least Upper Bounds for the Size of OBDDs Using Symmetry Properties. *IEEE Trans. on Comp.*, pages 360–368, Apr. 2000.

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

[5] S. Kannurao and B. J. Falkowski. Identification of Complement Single Variable Symmetry in Boolean Functions through Walsh Transform. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, pages V–745 – V–748, 2002.

[6] B. Kim and D. L. Dietmeyer. Multilevel Logic Synthesis of Symmetric Switching Functions. *IEEE Trans. on CAD*, pages 436–446, Apr. 1991.

[7] V. N. Kravets and K. A. Sakallah. Constructive Library-Aware Synthesis Using Symmetries. In *Proceedings of Design Automation and Test in Europe (DATE)*, pages 208–213, 2000.

[8] L. Litan, P. Molitor, and D. Möller. Least Upper Bounds on the Sizes of Symmetric Variable Order based OBDDs. In *Proceedings of the Great Lakes Symposium on VLSI*, pages 680–684, 1996.

[9] C. Meinel and T. Theobald. *Algorithms and Data Structures in VLSI Design*. Springer-Verlag, 1998.

[10] D. M. Miller. Spectral Symmetry Tests. In *Proceedings of the 11th International Symposium on Multiple-Valued Logic (ISMVL)*, pages 130–134, 1981.

[11] D. M. Miller and N. Muranaka. Multiple-Valued Decision Diagrams with Symmetric Variable Nodes. In *Proceedings of the 26th International Symposium on Multiple-Valued Logic (ISMVL)*, pages 242–247, 1996.

[12] C. Moraga. Complex Spectral Logic. In *Proceedings of the 8th International Symposium on Multiple-Valued Logic (ISMVL)*, pages 149–157, 1978.

[13] J. E. Rice and J. C. Muzio. Antiymmetries in the Realization of Boolean Functions. In *Proceedings of the International Symposium on Circuits and Systems (ISCAS)*, pages IV–69 – IV–72, 2002.

[14] M. A. Thornton and V. S. S. Nair. Efficient Calculation of Spectral Coefficients and Their Applications. *IEEE Trans. on CAD*, pages 1328 – 1341, Nov 1995.

[15] A. M. Tomaszewska, P. Dziurzanski, S. N. Yanushkevich, and V. P. Shmerko. Two-Stage Exact Detection of Symmetries. In *Proceedings of the 31st International Symposium on Multiple-Valued Logic (ISMVL)*, pages 213–216, 2001.

[16] C.-C. Tsai and M. Marek-Sadowska. Generalized Reed-Muller Forms as a Tool to Detect Symmetries. *IEEE Trans. on Comp.*, pages 33–40, Jan. 1996.