The 47th IEEE International Midwest
Symposium on Circuits and Systems

Generalization of Arithmetic-Haar Transform for Higher Dimensions

Bogdan J. Falkowski and Shixing Yan
School of Electrical and Electronic Engineering
Nanyang Technological University
Block S1, 50 Nanyang Avenue, Singapore 639798

***Abstract*—In this article, we extended arithmetic-Haar transform from the first eight functions (n = 3) to higher values of n. The new recursive relations are given in the form of layered Kronecker matrices and hence they have fast transforms and are computationally advantageous. As the new generalized arithmetic-Haar transform has a structure similar to that of the Haar and arithmetic transform matrices, computational advantage of these two transforms are held in the expanded transform as well.**

I. Introduction

Both the Haar wavelet transform (non-normalized version of the transform where only signs are entered into the transform matrix) and arithmetic transform have been used in many applications of logic design [1]–[4]. As each of these transforms has same advantages and disadvantages it is also beneficial to calculate the spectrum of a logic function by means of some other known spectrum of the same function without needing to regain the original function. Such a conversion for arithmetic and Haar spectra for arbitrary n were shown in [5]. In [6] an idea of a combined arithmetic-Haar transform was proposed. Such a transform was defined for the first eight functions and experimental results shown in [6] proved that arithmetic-Haar transform is more efficient than other used transforms in logic design such as Walsh, Haar and arithmetic for some benchmark functions. Therefore it is interesting not only theoretically but also practically, to develop this arithmetic-Haar transform for higher matrix dimensions and this is the main contribution of our article.

II. Arithmetic-Haar Transform for Three Variables

For a 3-variable function $f(x_1, x_2, x_3)$, the arithmetic-Haar expansions are given by the symbolic matrix [6]:

$$X = [1 \quad x_3 \quad x_1 \quad x_1x_3 \quad \overline{x}_1\overline{x}_3(1-2x_2) \quad \overline{x}_1x_3(1-2x_2) \quad x_1\overline{x}_3(1-2x_2) \quad x_1x_3(1-2x_2)]. \qquad (1)$$

Let the symbol '$\otimes$' represent Kronecker product of two matrices. The basic functions of arithmetic-Haar expansions can be combined from two sets of basic functions. The first four basic functions are generated from the positive Davio expansion [4] for variables x_1 and x_3:

$$[1 \quad x_1] \otimes [1 \quad x_3] = [1 \quad x_3 \quad x_1 \quad x_1x_3]. \qquad (2)$$

The other four basic functions are generated from the Shannon expansion [4] for variables x_1 and x_3, with multiplication by $(1-2x_2)$ [6]:

$$\{[\overline{x}_1 \quad x_1] \otimes [\overline{x}_3 \quad x_3]\} \times (1-2x_2) = [\overline{x}_1\overline{x}_3(1-2x_2) \quad \overline{x}_1x_3(1-2x_2) \quad x_1\overline{x}_3(1-2x_2) \quad x_1x_3(1-2x_2)]. \qquad (3)$$

III. Arithmetic-Haar Transform for Higher Number of Variables

A. Definition of Generalized Arithmetic-Haar Transform

For an n-variable function $f(x_1, x_2, \cdots x_r, \cdots x_{n-1}, x_n)$, the basic functions for the generalized arithmetic-Haar expansions can be combined from two sets of basic functions. The first 2^{n-1} basic functions are generated from the positive Davio expansion for variables x_1 to x_n, excluding x_r where $1 \leq r \leq n$:

$$[1 \quad x_1] \otimes [1 \quad x_2] \otimes \cdots \otimes [1 \quad x_{r-1}] \otimes [1 \quad x_{r+1}] \otimes \cdots \otimes [1 \quad x_{n-1}] \otimes [1 \quad x_n]. \qquad (4)$$

For the other 2^{n-1} basic functions, the functions can be generated from multiplying the Shannon expansion for variables x_1 to x_n, excluding x_r where $1 \leq r \leq n$:

$$[\overline{x}_1 \quad x_1] \otimes [\overline{x}_2 \quad x_2] \otimes \cdots \otimes [\overline{x}_{r-1} \quad x_{r-1}] \otimes [\overline{x}_{r+1} \quad x_{r+1}] \otimes \cdots \otimes [\overline{x}_{n-1} \quad x_{n-1}] \otimes [\overline{x}_n \quad x_n] \text{ by } (1-2x_r). \qquad (5)$$

Such an approach allows us to generalize the higher dimensions of final arithmetic-Haar matrices and corresponding expansions in many ways by selecting the final r, so this is a general method that can provide compact spectral representation with many zeros for any n-variable logic function.

Definition 1 From the generalized arithmetic-Haar expansions, the rth-order generalized arithmetic-Haar transform matrix $AH_r(n)$ and its inverse $AH_r^{-1}(n)$ can be defined as:

$$AH_r(n) = \begin{bmatrix} \left(\bigotimes_{i=1}^{r-1} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}\right) \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \left(\bigotimes_{i=1}^{n-r} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}\right) & \vdots \\ \left(\bigotimes_{i=1}^{r-1} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) \otimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \otimes \left(\bigotimes_{i=1}^{n-r} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) & \end{bmatrix}, \qquad (6)$$

$$AH_r^{-1}(n) = \frac{1}{2} \begin{bmatrix} \left(\bigotimes_{i=1}^{r-1} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}\right) \otimes [1 \quad 1] \otimes \left(\bigotimes_{i=1}^{n-r} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}\right) \\ \hline \left(\bigotimes_{i=1}^{r-1} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) \otimes [1 \quad -1] \otimes \left(\bigotimes_{i=1}^{n-r} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) \end{bmatrix} \qquad (7)$$

where $n = 2, 3, 4 \cdots$ and $1 \leq r \leq n$.

In the above equations, the symbol '$\bigotimes_{i=1}^{j}$' represents the Kronecker product of j matrices. When the Kronecker product of j matrices is carried out for the above equations for $j = 0$, then the term '$\bigotimes_{i=1}^{j}$' disappears from the above equations. The vertical dotted lines denote the layered vertical Kronecker matrices, and the horizontal dashed lines denote the layered horizontal Kronecker matrices, respectively. The layered horizontal Kronecker matrix is defined as the horizontal sum of Kronecker matrices [7], and the layered vertical Kronecker matrix is defined as the vertical sum of Kronecker matrices [5].

Definition 1 has shown the generation of both the forward and inverse generalized arithmetic-Haar transform. Using (6) and (7) the corresponding generalized arithmetic-Haar transform matrix can be calculated.

Example 1 For $n = 4$ and $r = 3$, the generalized arithmetic-Haar forward transform matrix $AH_3(4)$ can be generated by (6) as follow:

$$AH_3(4)=\left[\begin{array}{c:c} \left(\bigotimes_{i=1}^{2}\begin{bmatrix}1&0\\1&1\end{bmatrix}\right)\otimes\begin{bmatrix}1\\1\end{bmatrix}\otimes\begin{bmatrix}1&0\\1&1\end{bmatrix} & \left(\bigotimes_{i=1}^{2}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\otimes\begin{bmatrix}1\\-1\end{bmatrix}\otimes\begin{bmatrix}1&0\\0&1\end{bmatrix}\end{array}\right]=$$

$$\begin{bmatrix}
1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&-1&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&-1&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&0&0&1&0&0&0&0&0\\
1&1&1&1&0&0&0&0&0&0&0&1&0&0&0&0\\
1&0&1&0&0&0&0&0&0&0&-1&0&0&0&0&0\\
1&1&1&1&0&0&0&0&0&0&0&-1&0&0&0&0\\
1&0&0&0&1&0&0&0&0&0&0&0&1&0&0&0\\
1&1&0&0&1&1&0&0&0&0&0&0&0&1&0&0\\
1&0&0&0&1&0&0&0&0&0&0&0&-1&0&0&0\\
1&1&0&0&1&1&0&0&0&0&0&0&0&-1&0&0\\
1&0&1&0&1&0&1&0&0&0&0&0&0&0&1&0\\
1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&1\\
1&0&1&0&1&0&1&0&0&0&0&0&0&0&-1&0\\
1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&-1
\end{bmatrix},$$

Similarly, its inverse $AH_3^{-1}(4)$ can be calculated by (7) as follow:

$$AH_3^{-1}(4)=\frac{1}{2}\left[\begin{array}{c}\left(\bigotimes_{i=1}^{2}\begin{bmatrix}1&0\\-1&1\end{bmatrix}\right)\otimes\begin{bmatrix}1&1\end{bmatrix}\otimes\begin{bmatrix}1&0\\-1&1\end{bmatrix}\\ \hdashline \left(\bigotimes_{i=1}^{2}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\otimes\begin{bmatrix}1&-1\end{bmatrix}\otimes\begin{bmatrix}1&0\\0&1\end{bmatrix}\end{array}\right]=$$

$$\frac{1}{2}\begin{bmatrix}
1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
-1&1&-1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
-1&0&-1&0&1&0&1&0&0&0&0&0&0&0&0&0\\
1&-1&1&-1&-1&1&-1&1&0&0&0&0&0&0&0&0\\
-1&0&-1&0&0&0&0&0&1&0&1&0&0&0&0&0\\
1&-1&1&-1&0&0&0&0&-1&1&-1&1&0&0&0&0\\
1&0&1&0&-1&0&-1&0&-1&0&-1&0&1&0&1&0\\
-1&1&-1&1&1&-1&1&-1&1&-1&1&-1&-1&1&-1&1\\
1&0&-1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&-1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&-1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&-1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&-1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&-1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&-1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&-1
\end{bmatrix}.$$

Comparatively, the most known transform which has been applied to many areas including logic synthesis and optimization is Walsh-Hadamard transform [2]-[4], [7]. Similarly, the new arithmetic-Haar transform can also be used for the logic design.

Example 2 Consider a 4-variable function $f_1(x_1, x_2, x_3, x_4) = \sum(3, 5, 6, 7, 11, 12, 13, 14, 15)$. Using the generalized arithmetic-Haar transform in Example 1 and Walsh-Hadamard transform for f_1, the corresponding arithmetic-Haar coefficients and Walsh-Hadamard coefficients can be calculated respectively as follow:

$$AH_3^{-1}(4)\times\begin{bmatrix}0\\0\\0\\1\\0\\1\\1\\1\\0\\0\\0\\1\\1\\1\\1\\1\end{bmatrix}=\frac{1}{2}\begin{bmatrix}0\\1\\1\\0\\0\\1\\-1\\0\\-1\\-1\\0\\0\\-1\\0\\0\end{bmatrix},\ \left(\bigotimes_{i=1}^{4}\begin{bmatrix}1&1\\1&-1\end{bmatrix}\right)^{-1}\times\begin{bmatrix}0\\0\\0\\1\\0\\1\\1\\1\\0\\0\\0\\1\\1\\1\\1\\1\end{bmatrix}=\frac{1}{16}\begin{bmatrix}9\\-3\\-3\\1\\-5\\-1\\-1\\3\\-1\\-1\\-1\\-1\\1\\1\\1\\1\end{bmatrix}.$$

Using Eqs. 4 and 5 the corresponding arithmetic-Haar expansion for the function f_1 is:
$f_1 = \frac{1}{2}[x_2 + x_4 + x_1x_2 - x_1x_2x_4 - \bar{x}_1\bar{x}_2x_4(1 - 2x_3) - \bar{x}_1x_2\bar{x}_4(1 - 2x_3) - x_1\bar{x}_2x_4(1 - 2x_3)]$.

From the Walsh-Hadamard coefficients calculated above, the corresponding Walsh-Hadamard polynomial expansion for the function f_1 is:
$f_1 = \frac{1}{16}[9 - 3(1 - 2x_4) - 3(1 - 2x_3) + (1 - 2x_3)(1 - 2x_4) - 5(1 - 2x_2) - (1 - 2x_2)(1 - 2x_4) - (1 - 2x_2)(1 - 2x_3) + 3(1 - 2x_2)(1 - 2x_3)(1 - 2x_4) - (1 - 2x_1) - (1 - 2x_1)(1 - 2x_4) - (1 - 2x_1)(1 - 2x_3) - (1 - 2x_1)(1 - 2x_3)(1 - 2x_4) + (1 - 2x_1)(1 - 2x_2) + (1 - 2x_1)(1 - 2x_2)(1 - 2x_4) + (1 - 2x_1)(1 - 2x_2)(1 - 2x_3) + (1 - 2x_1)(1 - 2x_2)(1 - 2x_3)(1 - 2x_4)]$.

In this example, it is obvious that the arithmetic-Haar coefficients have less non-zero coefficients than the Walsh-Hadamard coefficients, so the arithmetic-Haar transform and the corresponding polynomial expansion are more efficient than the Walsh-Hadamard transform and its expansion.

B. Fast Algorithms and Computational Costs

Since the generalized arithmetic-Haar transform has the recursive relations given in the form of layered Kronecker matrices, it is possible to derive fast algorithm for the calculation of the generalized arithmetic-Haar transform matrices. Similar to the fast Walsh transform and other known fast transforms, the generalized arithmetic-Haar transform matrix can be calculated by the products of the factored matrices. The fast algorithms for the forward and inverse generalized arithmetic-Haar transforms are presented in the following properties.

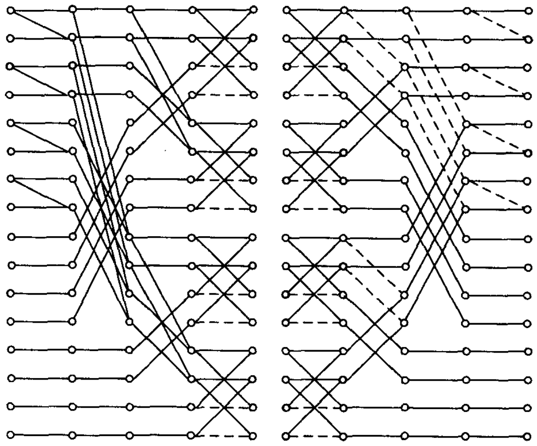

Fig. 1. Fast butterfly diagram for (a) forward and (b) inverse generalized arithmetic-Haar transforms, $r = 3$ and $n = 4$.

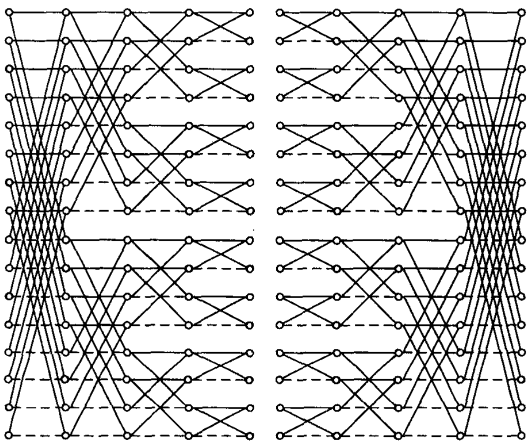

Fig. 2. Fast butterfly diagram for (a) forward and (b) inverse Walsh-Hadamard transforms where $n = 4$.

Property 1 The forward generalized arithmetic-Haar transform $AH_r(n)$ can be factorized as:

$$AH_r(n) = \prod_{i=1}^{n} D_r^i(n) \tag{8}$$

where

$$D_r^i(n)=\begin{cases} \left(\bigotimes_{j=1}^{r-1}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\otimes\begin{bmatrix}1&1\\1&-1\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{n-r}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right) & \text{for } i=1,\\ \left(\bigotimes_{j=1}^{r-i}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\otimes\begin{bmatrix}1&0&0&0\\0&0&1&0\\1&1&0&0\\0&0&0&1\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{n+i-r-2}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right) & \text{for } 1<i\leq r,\\ \begin{bmatrix}\begin{bmatrix}1&0\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{i-2}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\otimes\begin{bmatrix}1&0\\1&1\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{n-i}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\\ \hdashline \begin{bmatrix}0&1\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{n-1}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\end{bmatrix} & \text{for } r<i\leq n.\end{cases}$$

Property 2 The inverse generalized arithmetic-Haar transform $AH_r^{-1}(n)$ can be factorized as:

$$AH_r^{-1}(n) = \frac{1}{2}\prod_{i=1}^{n} E_r^i(n) \tag{9}$$

where $E_r^i(n)=$

$$\begin{cases} \begin{bmatrix}\begin{bmatrix}1\\0\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{n-i-1}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\otimes\begin{bmatrix}1&0\\-1&1\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{i-1}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right) \vdots \quad \begin{bmatrix}0\\1\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{n-1}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\end{bmatrix} & \text{for } 1\leq i<n-r+1,\\ \left(\bigotimes_{j=1}^{r+i-n-1}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\otimes\begin{bmatrix}1&0&0&0\\-1&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{2n-i-r-1}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right) & \text{for } n-r+1\leq i<n,\\ \left(\bigotimes_{j=1}^{r-1}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)\otimes\begin{bmatrix}1&1\\1&-1\end{bmatrix}\otimes\left(\bigotimes_{j=1}^{n-r}\begin{bmatrix}1&0\\0&1\end{bmatrix}\right) & \text{for } i=n.\end{cases}$$

Such fast algorithms will greatly reduce the number of arithmetic operations as compared to the computation of generalized arithmetic-Haar transform by the whole matrix. Using this approach, the computational costs for all the cases of the generalized arithmetic-Haar transform are $(n+3)2^{n-1}$, where the computational costs mean the number of additions and subtractions required for the generation of forward and inverse transforms.

Example 3 From Property 1, $AH_3(4)$ in Example 1 can be factorized using (8). Hence, $AH_3(4)$ can be expressed as the product of four matrices as follows:

$$AH_3(4) = \begin{bmatrix}
1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&-1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&-1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&-1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&-1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&-1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&-1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&-1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&-1
\end{bmatrix} \times$$

$$\begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&1&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1
\end{bmatrix}$$

$$
\times \begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0\\
1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1
\end{bmatrix}
\times \begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1
\end{bmatrix}.
$$

Similarly, the inverse generalized arithmetic-Haar transform $AH_3^{-1}(4)$ in Example 1 can be also factorized using (9). From the representation of $AH_3(4)$ and $AH_3^{-1}(4)$ by products of the factored matrices, the corresponding fast flow diagrams for the calculation of forward and inverse generalized arithmetic-Haar transform matrices can be drawn as shown in Fig. 1 (a) and (b), accordingly. In all the figures in this article, the solid lines and dotted lines represent addition and subtraction, respectively. In comparison with the generalized arithmetic-Haar transform, the fast diagrams for calculation of forward and inverse Walsh-Hadamard transforms for 4-variable functions are shown in Fig. 2 (a) and (b), accordingly [4], [7]. Comparing Fig. 1 and Fig. 2, the computational advantages of the generalized arithmetic-Haar transform over known Walsh-Hadamard transform are clearly seen as the total number of operations required to perform the new transform are less than the ones for Walsh-Hadamard transform.

The computational costs of generalized arithmetic-Haar transform have been discussed in this section, and the computational costs of Walsh transform and arithmetic transform are known as $n2^{n+1}$ and $n2^n$, respectively. Consequently, the computational costs of generalized arithmetic-Haar transform, Walsh transform and arithmetic transform can be compared in detail. Table I shows that the computational costs of Walsh and arithmetic transforms increase considerably when compared with the generalized arithmetic-Haar transform for higher n. This shows the computational advantages of generalized arithmetic-Haar transform over Walsh and arithmetic transforms.

IV. Conclusion

In this paper, new relations for higher dimensions of generalized arithmetic-Haar transform are presented. Since the given equations are very general, they can develop the whole family of generalized arithmetic-Haar transform, where each of the member of this family has a little bit different structure and properties. Such an approach will allow to use the best arithmetic-Haar transform for a given logic function in different situations. The fast algorithms of generalized arithmetic-Haar transform are also introduced in this article, which provide the efficient way to calculate the spectrum. In comparison with known Walsh and arithmetic transforms, generalized arithmetic-Haar transform showed the computational advantages over the other two transforms.

While the application of generalized arithmetic-Haar transform in this paper is focused on logic design, the presented derivations may be also useful for applications of generalized arithmetic-Haar transform in areas others than logic design, for example in digital signal and image processing.

References

[1] B. J. Falkowski, "A note on the polynomial form of Boolean functions and related topics," *IEEE Trans. on Computers*, vol. 48, no. 8, pp. 860–864, Sept. 1999.

[2] M. G. Karpovsky, *Finite Orthogonal Series in the Design of Digital Devices*, John Wiley, New York, 1976.

[3] M. G. Karpovsky, R. S. Stankovic, and J. T. Astola, "Spectral techniques for design and testing of computer hardware," in *Proc. 1st Int. Workshop on Spectral Techniques and Logic Design for Future Digital Systems*, Tampere, Finland, June 2000, pp. 9–42, published as TICSP Series #10, Dec. 2000.

[4] R. S. Stankovic, M. S. Stankovic, and D. Jankovic, *Spectral Transforms in Switching Theory, Definitions and Calculations*, Science Publisher, Belgrade, 1998.

[5] B. J. Falkowski, "Relationship between arithmetic and Haar wavelet transforms in the form of layered Kronecker matrices," *Electronics Letters*, vol. 35, no. 10, pp. 799–800, May 1999.

[6] R. S. Stankovic, C. Moraga, and J. T. Astola, "From Fourier expansions to arithmetic-Haar expressions on quaternion groups," *Applicable Algebra in Engineering, Communication and Computing*, vol. 12, no. 3, pp. 227–253, 2001.

[7] L. P. Yaroslavsky, *Digital Picture Processing - An Introduction*, Springer-Verlag, Berlin, 1985.

TABLE I
Comparison of Computational Costs

n	arithmetic-Haar $(n+3)2^{n-1}$	Walsh $n2^{n+1}$	arithmetic $n2^n$
2	10	16	8
3	24	48	24
4	56	128	64
5	128	320	160
6	288	768	384
7	640	1792	896
8	1408	4096	2048
9	3072	9216	4608
10	6656	20480	10240
11	14336	45056	22528
12	30720	98304	49152
13	65536	212992	106496
14	139264	458752	229376
15	294912	983040	491520
16	622592	2097152	1048576