Walsh-like functions and their relations

B.J. Falkowski
S. Rahardja

Indexing terms: Discrete transforms, Walsh functions, Haar functions

Abstract: A new discrete transform, the 'Haar–Walsh transform', has been introduced. Similar to well known Walsh and non-normalised Haar transforms, the new transform assumes only +1 and –1 values, hence it is a Walsh-like function and can be used in different applications of digital signal and image processing. In particular, it is extremely well suited to the processing of two-valued binary logic signals. Besides being a discrete transform on its own, the proposed transform can also convert Haar and Walsh spectra uniquely between themselves. Besides the fast algorithm that can be implemented in the form of in-place flexible architecture, the new transform may be conveniently calculated using recursive definitions of a new type of matrix, a 'generator matrix'. The latter matrix can also be used to calculate some chosen Haar–Walsh spectral coefficients which is a useful feature in applications of the new transform in logic synthesis.

1 Introduction

Discrete orthogonal Walsh-like transforms such as the Walsh transform, the Haar transform (HT), and others like the discrete-cosine transform, slant transform and Fourier transform, have been used in image processing, speech processing, pattern recognition and communication systems [1–7]. Each of these orthogonal transforms have advantages and disadvantages for various applications. Walsh and non-normalised Haar [4, 5, 8, 9] (rationalised [2]) orthogonal functions assume values +1 and –1 only, and with such, the computation of the transforms requires arithmetic addition and subtraction. Owing to this fact, they yield the most economical computational costs and have been used efficiently in applications involving two or more valued signals, situations that happen in digital signal processing of logical signals and in the design of multirate digital signal processing systems [1, 3–10].

Two modified transforms related to Walsh functions have been known: rapid and Hadamard–Haar transforms [2, 5]. The flow graph for the rapid transform is identical to that of the fast Walsh–Hadamard transform except that the absolute value of the output is taken before applying to the next stage. However, the rapid transform is nonorthogonal and does not have an inverse. Another modified transform is based on a hybrid version of the Haar and Walsh transforms [2, 11]. This transform is derived from different linear combinations of the basis Haar functions with an appropriate scaling factor. Such a combination of basis functions have been found advantageous for feature selection and pattern recognition. The rationalised version of this transform has also been introduced [12]. Another family of orthogonal functions related to the Walsh and Haar functions has been introduced [13–15]. They are called bridge functions and can be generated by copy theory originated from work by Swick [16] for Walsh functions. With the introduction of the broad family of bridge functions, both Haar and Walsh functions can be considered as the members of the same group. Similarly to our transform, the bridge functions share the good properties of both Haar and Walsh systems.

Many authors have considered the mutual relationships between Haar and Walsh functions. Kaczmarz and Steinhaus [17] defined Walsh functions in Paley order through the linear combination of normalised Haar functions by means of recursive equations. The same relations in the form of universal recursive equations operating on submatrices which translate Walsh–Paley functions to Haar functions and vice versa have been given in [7, 8]. Walsh functions in Kaczmarz ordering can also be generated through linear combinations of Haar functions. Such forward and inverse relations for Kaczmarz ordering in a submatrix form obtained through Walsh–Paley transforms with reorderings have been shown [18, 19]. Ways of obtaining Haar functions row by row from three Walsh orderings Hadamard, Kaczmarz and Paley [19] and also from Rademacher functions [9] have been developed. From relations between the submatrices of Haar and Walsh–Kaczmarz transformations (named wrongly in [18] as Walsh–Hadamard transforms), Fino [18] showed a simultaneous calculation of Haar and Walsh–Kaczmarz spectral coefficients with the aid of reorderings. However, the presented method is unable to calculate a single spectral coefficient.

In this paper, a transform that maps the Walsh–Kaczmarz spectrum into the Haar spectrum and vice versa is introduced. The new transform is based on the combination of Walsh and Haar basis functions. The new Haar–Walsh transform (HWT) is unique and has all the advantages of known Haar and Walsh transforms. The fast HWT may be simply derived by removing the fast Haar transform from the butterfly of the fast Walsh–Paley transform. However, the transform is

IEE Proceedings online no. 19960760
Paper first received 12th January and in revised form 26th June 1996

The authors are with Nanyang Technological University, School of Electrical and Electronic Engineering, Nanyang Avenue, Singapore 639798

unique and may operate directly not only on Walsh and Haar spectra but also on arbitrary binary/ternary data. Moreover, with the help of the novel generator matrix introduced in this paper, one is able to calculate a single spectral coefficient of fast HWT. This is in high contrast to previous developments [5, 8, 17–19]. The HWT may be used in applications where traditional Haar and Walsh transforms have been used, for example in image processing and digital signal processing of logical signals.

There are a few reasons for introducing this transform. First, besides having all the advantages of existing discrete transforms, the new transform can serve another purpose: the forward HWT can transform spectral data from the Haar domain to the Walsh–Paley domain, and its inverse does exactly the opposite. The important application of the new transform makes all three spectral representations available at any stage of the digital signal processing design process, thus allowing an engineer to choose the form that is most suitable for a given application. An efficient way of calculating the transform, using the generator matrix concept has also been developed. Different new operators on generator matrices as well as evaluation of the complexity calculating the HWT by using such matrices have been provided. The existence of a fast flow diagram has substantially reduced the complexity of the new transform. Another important property of the new transform is that the fast flow diagram of the forward HWT and inverse HWT are identical. Furthermore, the implementation of the new transform in hardware is possible. The structure permits in-place architecture which reduces components' requirements.

2 Basic definitions of Walsh-like functions

In many applications of Walsh-like functions, representations of the orthogonal functions in matrix form are generally preferred. There are four commonly cited Walsh orderings: Hadamard, Kaczmarz, Rademacher and Paley [2, 4–7, 9, 15, 20, 21]. For the Walsh–Kaczmarz [4, 5, 17] transformation matrix, they are usually generated by reordering the rows of a class of Hadamard matrices [1, 2, 4–9, 20–22], which are known to be conveniently generated by Kronecker products. Let $WH_{[N]}$ and $H_{[N]}$ be $N \times N$ matrices defining Walsh–Hadamard and the normalised Haar transforms accordingly, and $N = 2^n$. Then [1, 3–9],

$$WH_{[N]} = \begin{bmatrix} WH_{[\frac{N}{2}]} & WH_{[\frac{N}{2}]} \\ WH_{[\frac{N}{2}]} & -WH_{[\frac{N}{2}]} \end{bmatrix} \tag{1}$$

$$H_{[N]} = \begin{bmatrix} H_{[\frac{N}{2}]} \otimes [1 \quad 1] \\ I_{[\frac{N}{2}]} \otimes [1 \quad -1] \end{bmatrix} \tag{2}$$

where $WH_{[1]} = H_{[1]} = [1]$. Let $WK_{[N]}$ be defined as the Walsh–Kaczmarz transform. Then, by reordering the rows in $WH_{[N]}$, $WK_{[N]}$ is derived. Alternatively, the Walsh–Kaczmarz matrix may be derived from the set of N discrete Walsh–Rademacher functions. Various ways of effectively generating Walsh functions in different orderings have been proposed and mutual relationships between them have been investigated [1, 3–9, 15–17, 20–23].

Let $f(j)$ denote a data sequence with $0 \le j \le N-1$. Let the column vector $\mathbf{f}$ denote the elements of the data sequence $f(j)$, where $\mathbf{f} = [f(0), f(1), ..., f(j), ..., f(N-1)]^T$. Then, the corresponding Walsh–Kaczmarz transform of the data sequence may be expressed in the matrix form

$$\mathbf{F_{WK}} = WK_{[N]}\mathbf{f} \tag{3}$$

where $WK_{[N]}$ is the Walsh–Kaczmarz transformation matrix of order N.

We refer to the transformed data sequence as the discrete transformation's spectrum, and its elements are known as the spectral coefficients. If $WK_{[N]}$ in eqn. 3 is replaced by $WP_{[N]}$ or $H_{[N]}$, then the resulting transformed vector, denoted as $\mathbf{F_{WP}}$ or $\mathbf{F_H}$, will contain the Walsh–Paley and Haar spectral coefficients, respectively. The Haar–Walsh transform (HWT) is derived from the combination of Walsh and Haar functions. Let $T_{[N]}$ denote the HWT matrix of dimension $N \times N$, where $N = 2^n$. Then the forward HWT matrix $T_{[N]}$ is defined as

$$T_{[N]} = \begin{bmatrix} T_{[\frac{N}{2}]} & 0 \\ 0 & WP_{[\frac{N}{2}]} \end{bmatrix} \tag{4}$$

where $T_{[1]} = WP_{[1]} = 1$. $T_{[N]} = T_{[N]}^{-1}$ when $T_{[N]}^{-1}$ is made non-normalised. It should be noted that for non-normalised basis functions the matrix $T_{[N]}$ can be applied directly to non-normalised Haar functions, and in such a case, the single matrix $T_{[N]}$ describes the same relationship between basis Haar and Walsh functions which was given by the set of recursive equations. These equations were first illustrated by Kaczmarz [17] and also through relations on submatrices [5, 8, 18, 21].

3 Definition and properties of generator matrix

In this section, the basic definitions and properties of a generator matrix are presented. New matrix operations useful for the computation of the generator matrix are introduced. The generator matrix is used in the computation of spectral coefficients of the HWT. It is also possible to compute a chosen spectral coefficient of the transform. We define $G^{\langle m \rangle}$ as a generator matrix, which is a rectangular matrix of dimension $2^{m-1} \times 2^{m-1}-1$ with elements $\{1, -1\}$ represented as $\{+, -\}$ respectively, such that the scalar product of any two columns of $G^{\langle m \rangle}$ is either 0 or 2^m. The basis vector of the generator matrix denoted as $\mathbf{u}$ is a 2×1 column vector defined as

$$\mathbf{u} = \begin{bmatrix} + \\ - \end{bmatrix}$$

The generator matrix $G^{\langle m \rangle}$ has a recursive structure

$$G^{\langle m \rangle} = \left[G^{\langle m-1 \rangle} \bullet 2 \,\vdots\, \mathbf{u} \circ 2^{m-2} \,\vdots\, G^{\langle m-1 \rangle} \bullet 2 \right] \tag{5}$$

where $m \ge 2$, and the matrix operators $\circ$ and $\bullet$ are defined as follows. Let A be an $r \times c$ matrix. The matrix operator $\circ$ of the matrix A with a scalar k is the partitioned matrix B of dimension $rk \times c$ such that its first r rows are exactly the same as the rows of matrix A, the rows from $r+1$ to $2r$ are again exactly the same as the rows of matrix A, etc. Hence,

$$A \circ k = B = [[A], [A], \overset{k \text{ times}}{\ldots}, [A]]^T \tag{6}$$

Let A be an $r \times c$ matrix, where $A = [\mathbf{R}_1, \mathbf{R}_2, ..., \mathbf{R}_r]^T$. The matrix operator $\bullet$ of the matrix A with a scalar k is the partitioned matrix B of dimension $rk \times c$ such that its first k rows are exactly the same as the first rows of matrix A, the rows from $k+1$ to $2k$ are exactly the same as the second row of matrix A etc. Hence

$$A \bullet k = B = [[\mathbf{R_1}, \overset{k \text{ times}}{\ldots}, \mathbf{R_1}]^T, [\mathbf{R_2}, \overset{k \text{ times}}{\ldots}, \mathbf{R_2}]^T, \ldots, [\mathbf{R_r}, \overset{k \text{ times}}{\ldots}, \mathbf{R_r}]^T] \tag{7}$$

Example 1: Let

$$A = \begin{bmatrix} + & + \\ - & - \end{bmatrix}$$

Then

$$A \circ 2 = \begin{bmatrix} + & + \\ - & - \\ + & + \\ - & - \end{bmatrix} \quad A \bullet 2 = \begin{bmatrix} + & + \\ + & + \\ - & - \\ - & - \end{bmatrix}$$

respectively. It should be mentioned that only for k = 2, the RS operator used to generate the Walsh–Kaczmarz transformation matrix [5, 25] could be applied to A and would result in,

$$RS(A) = \begin{bmatrix} + & + & + & + \\ + & + & - & - \\ - & - & + & + \\ - & - & - & - \end{bmatrix} = [A \bullet 2 \mid A \circ 2]$$

Table 1 shows some generator matrices. When m = 1, the generator matrix becomes a null matrix $[\phi]$.

Table 1: Forward and inverse generator matrix table

m	$\mathbf{F_H}^{\langle m \rangle}$	Matrix, $\mathbf{G}^{\langle m \rangle}$	$\mathbf{F_{WP}}^{\langle m \rangle}$
1	$[F_H(0), F_H(1)]^T$	$[\phi]$	$[F_{WP}(0), F_{WP}(1)]^T$
2	$[F_H(2^{m-1}+v)]^T$, $0 \le v \le 2^{m-1}-1$	+ −	$[F_{WP}(2^{m-1}+v)]^T$, $0 \le v \le 2^{m-1}-1$
3		+++ +−+ −+− −−−	
4		+++++++ +++−+++ +−+++−+ +−+−+−+ −+−+−+− −+−−−+− −−−+−−− −−−−−−−	
...		...	

Example 2: When m = 2, the generator matrix

$$G^{\langle 2 \rangle} = \left[G^{\langle 2-1 \rangle} \bullet 2 \,\vdots\, \mathbf{u} \circ 2^{2-2} \,\vdots\, G^{\langle 2-1 \rangle} \bullet 2 \right]$$

$$\text{and } G^{\langle 2 \rangle} = \begin{bmatrix} + \\ - \end{bmatrix}$$

For m = 3, the generator matrix is

$$G^{\langle 3 \rangle} = \left[G^{\langle 3-1 \rangle} \bullet 2 \,\vdots\, \mathbf{u} \circ 2^{3-2} \,\vdots\, G^{\langle 3-1 \rangle} \bullet 2 \right]$$

$$= \left[G_F^{\langle 2 \rangle} \bullet 2 \,\vdots\, \mathbf{u} \circ 2 \,\vdots\, G^{\langle 2 \rangle} \bullet 2 \right]$$

$$= \left[\begin{bmatrix} + \\ - \end{bmatrix} \bullet 2 \,\vdots\, \begin{bmatrix} + \\ - \end{bmatrix} \circ 2 \,\vdots\, \begin{bmatrix} + \\ - \end{bmatrix} \bullet 2 \right]$$

$$= \begin{bmatrix} + & + & + \\ + & - & + \\ - & + & - \\ - & - & - \end{bmatrix}$$

The above definitions of matrix operators allow recursive computation of the generator matrix. There exists a direct mathematical relation between the Haar and Walsh–Paley spectral coefficients involving the generator matrix. Before this relation can be formulated, we first define the generator internal product $\diamond$.

If $\mathbf{F}_1 = [F_1(0), F_1(1), ..., F_1(2^m - 1)]^T$ is a $2^m \times 1$ column vector and the elements of generator matrix $G^{\langle m \rangle}$ are defined by $g(i, j)$, then their generator internal product $\mathbf{F}_2$ is

$$\mathbf{F_2} = G^{\langle m \rangle} \diamond \mathbf{F_1} = [F_2(0), F_2(1), \ldots, F_2(2^m - 1)]^T \quad (8)$$

which is defined as a $2^m \times 1$ vector whose entries $F_2(i)$ may be expressed as

$$F_2(i) = [\ldots[F_1(2^{m-1} - 2)g(i, 2^{m-1} - 2)F_1(2^{m-1} - 1)]$$
$$g(i, 2^{m-1} - 1)[F_1(2^{m-1})g(i, 2^{m-1})F_1(2^{m-1} + 1)]\ldots] \quad (9)$$

where $1 \le i \le 2^{m-1}$ and $1 \le j \le 2^{m-1} - 1$.

If A is a null generator matrix and $\mathbf{F}$ is a column vector with arbitrary dimension then the generator internal product is: $A \diamond \mathbf{F} = [\phi] \diamond = \mathbf{F} = \mathbf{F}$.

Let $M_{[N]}$ be an $N \times N$ ($N = 2^n$) square matrix such that $M_{[N]}$ is recursively defined. The rotation operator R on the square matrix $M_{[N]}$ is recursively defined as 4^{n-j} clockwise rotations involving 4^{n-j+1} submatrices each of order 2^{j-1} for $j = n, n-1, ..., 2, 1$.

4 Computation of Haar–Walsh transform using generator matrix

The basic definitions and properties of the generator matrix were introduced in Section 3. These new equations will be utilised to generate and calculate discrete transformations. The presented method also allows one to calculate a single spectral coefficient of the HWT. Before the generator internal product can be used to formulate the mathematical relations between the Haar and Walsh–Paley spectral coefficients, each of the transformed vectors is expressed as the combination of the corresponding subvectors $\mathbf{F_{WP}}^{\langle m \rangle}$ and $\mathbf{F_H}^{\langle m \rangle}$, whose elements are chosen from Walsh–Paley and Haar spectral coefficients in a manner described by equations in Table 1. If n is the number of variables, then the Walsh–Paley and Haar spectra ($\mathbf{F_{WP}}$ and $\mathbf{F_H}$) have elements derived by superposition of the elements in all the subvectors $\mathbf{F_{WP}}^{\langle m \rangle}$ and $\mathbf{F_H}^{\langle m \rangle}$ obtained recursively from m = 1 to m = n. As mentioned earlier, the HWT serves as a transform to convert the Haar spectra into the Walsh–Paley spectra and vice versa. However, the transform is able to stand on its own, as another discrete transformation like the Haar and Walsh–Paley transforms.

Let the $2^n \times 1$ column vector $\mathbf{F_H}$ be either the truth vector of an n-variable function or its Haar spectrum. Let $2^n \times 1$ column vector $\mathbf{F_{WP}}$ be the spectrum of the vector $\mathbf{F_H}$ for the HWT.

Algorithm 1: calculation of forward and inverse HWT spectra

(1) Apply equations and generator matrix G shown in Table 1 to generate vector $\mathbf{F_{WP}}$ defined as

$$\mathbf{F_{WP}} = G \diamond \mathbf{F_H} \quad (10)$$

composed of the respective subvectors

$$\mathbf{F}_{\mathbf{WP}}^{\langle \mathbf{m} \rangle} = G^{\langle m \rangle} \diamond \mathbf{F}_{\mathbf{H}}^{\langle \mathbf{m} \rangle} \quad (11)$$

for $\forall\, m$, $1 \le m \le n$.

(2) Apply equations and generator matrix G shown in Table 1 to generate vector $\mathbf{F_H}$ defined as

$$\mathbf{F_H} = G \diamond \mathbf{F_{WP}} \quad (12)$$

composed of the respective subvectors

$$\mathbf{F}_{\mathbf{H}}^{\langle \mathbf{m} \rangle} = 2^{1-m} \left(G^{\langle m \rangle} \diamond \mathbf{F}_{\mathbf{WP}}^{\langle \mathbf{m} \rangle} \right) \quad (13)$$

for $\forall\, m$, $1 \le m \le n$.

5 Fast Haar–Walsh transform

The existence of a fast flow diagram in a discrete transformation is essential as it facilitates the computation of the transform and reduces costs tremendously. Among the discrete transformations, the Haar transform is known to have the most efficient computational cost. Walsh and discrete Fourier transforms have equal computational cost, but the former requires no complex operations and its function values assume only +1 and –1. In comparison to the Haar transform, the Walsh transform has a higher computational cost, but using only the first stage of an N-point Haar forward fast flow diagram and repeating it ($\log_2 N$) times, an in-place constant architecture for the Walsh–Hadamard transform is feasible [1, 3–5, 20].

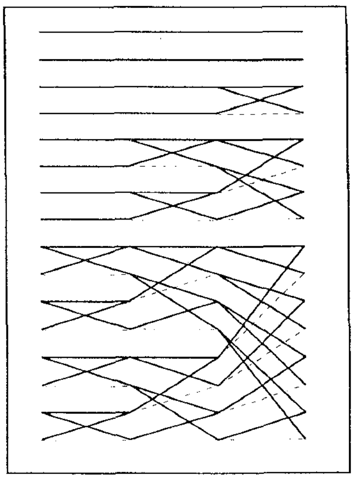

Fig. 1 *Fast flow diagram of Haar–Walsh transform for N = 16*

Table 2 compares the computational costs of different discrete transformations. The computational cost of calculating the full HWT by the generator matrix is high, though it is convenient as it allows computation of any single Haar–Walsh Paley (HWP) spectral coefficient. It may be worth observing that this property does not apply for any other discrete transformations. However, when applications require full HWP spectral coefficients, the generator matrix will be highly unprofitable. The existence of a fast flow diagram for HWT is thus essential. Fig. 1 shows the fast flow diagram of HWT for $N = 16$. The HWT fast flow diagram may be easily constructed by observing its distinct property. The first-half of an N-point HWT fast flow diagram is matched with the $N/2$-point HWT fast flow diagram, duplicating twice, shifting one stage left and combining with the last stage of the $N/2$-point inverse Haar fast flow diagram, where by definition, when $N = 2$, no operations or computation are needed for the HWT. Alternatively, the N-point HWT fast flow diagram may be derived by connecting one two-point fast identity transform and ($\log_2 N - 1$) fast transforms in parallel, where $\log_2 N \in \mathbf{Z}$ and $N > 2$.

The HWT fast flow diagram has significant applications: first, the calculation of the Walsh–Paley transform by using its own fast flow diagram is economically identical to the process of first using the non-normalised Haar fast flow diagram and then the HWT fast flow diagram. However, the latter is more auspicious as it results in obtaining two spectra, i.e. Haar and Walsh–Paley spectra, allowing a designer to fully utilise the distinct advantages and properties of both the Haar and Walsh–Paley spectra. Secondly, in applications where the HWT stands itself as a discrete transformation, the same fast flow diagram may be used for both forward and inverse transformations. The HWT is very well suited for hardware architecture due to the earlier mentioned properties of the fast flow diagram. With such a structure, an in-place flexible parallel processing architecture could be developed to reduce the size of hardware requirements. Assuming that the hardware requirement is proportional to the number of additions and subtractions, and each addition (or subtraction) corresponds to the unit size of hardware, the size of hardware required to calculate forward or inverse HWT using in-place flexible switching parallel processing hardware architecture and a standard architecture are equal to $\Sigma_{m=1}^{n-1} 2^m = 2(2^{n-1} - 1)$ and $[(n-2)2^n + 2]$, respectively. The reduction in size will be equal to $((n-3)2^n + 4)/((n-2)2^n + 2)) \times 100\%$. Table 2 shows the possible reduction in hardware size of implementing HWT using in-place hardware architecture.

The HWT can be used for conversion between Haar and Walsh–Paley spectral coefficients. By permuting the resulting HWT spectra, the Walsh spectral coefficients in Hadamard and Kaczmarz ordering may be derived. The bit reversal algorithm [6] is used for obtaining the Walsh–Hadamard spectral coefficients. The following bit permutation algorithm is useful for obtaining the final Walsh–Kaczmarz spectra from the Haar spectra. Let BP define the $N \times N$ bit permutation transform matrix, then

$$BP_{[N]} = \begin{bmatrix} BP_{[\frac{N}{2}]} & 0 \\ 0 & R\left\{BP_{[\frac{N}{2}]}\right\} \end{bmatrix} \tag{14}$$

where $R\{BP_{[1/2\,N]}\}$ defines the rotation operation on the matrix BP, where BP is subdivided into four $1/4\ N \times 1/4\ N$ submatrices and each submatrix is grouped and

Table 2: Comparison of computational costs, $N = 2^n$

n	N	WHT	HT	HWT by generator matrix	Fast HWT	Reduction in hardware size (%)
		$N\log_2 N$	$2(N-1)$	$^2/_3 + {}^1/_3 N^2 - 2N$	$N(\log_2 N - 2) + 2$	$[(n-3)2^n+4]/[(n-2)2^n+2]$
1	2	2	2	0	0	–
2	4	8	6	2	2	0
3	8	24	14	14	10	40
4	16	64	30	70	34	58.82
5	32	160	62	310	98	69.39
6	64	384	126	1302	258	75.97
7	128	896	254	5334	642	80.37

rotated clockwise. Eqn. 14 shows that the *BP* algorithm can be implemented by a $(n - 1)$-stage fast flow diagram. Fig. 2 shows the flow diagram of the bit permutation algorithm for $N = 16$.

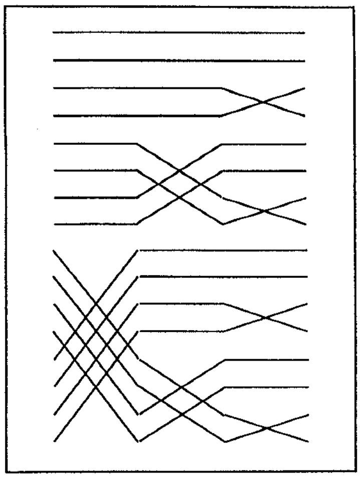

Fig.2 *Flow diagram of bit permutation algorithm for N = 16*

6 Example

Let $N = 8$ and the data vector $\mathbf{f} = [1, 0, 0, 1, 0, 1, 0, 1]^T$. For this data, the normalised Haar and Walsh–Hadamard spectral coefficients are $\mathbf{F_H} = [4, 0, 0, 0, 1, -1, -1, -1]^T$ and $\mathbf{F_{WH}} = [4, -2, 0, 2, 0, 2, 0, 2]^T$, respectively. For $N = 8$, the Haar–Walsh transformation matrix and bit permutation matrix may be derived from eqns. 4 and 14, respectively,

$$T_{[8]} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \end{bmatrix}$$

$$\text{and } BP_{[8]} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Then, the Walsh–Paley and Walsh–Kaczmarz spectral coefficients are

$$\mathbf{F_{WP}} = T_{[8]}\mathbf{F_H} = [4 \quad 0 \quad 0 \quad 0 \quad -2 \quad 2 \quad 2 \quad 2]^T$$

and

$$\mathbf{F_{WK}} = BP_{[8]}T_{[8]}\mathbf{F_H} = [4 \quad 0 \quad 0 \quad 0 \quad 2 \quad 2 \quad 2 \quad -2]^T$$

respectively.

Applying $\mathbf{F_H}$ into a fast flow diagram, Walsh–Paley spectral coefficients are derived. Using the bit permutation algorithm and the bit reversal algorithm [6]

$$BR = \prod_{p=1}^{n-1} I_{[2^{n-p-1}]} \otimes \begin{bmatrix} I_{[2^p]} \otimes [1 \quad 0] \\ I_{[2^p]} \otimes [0 \quad 1] \end{bmatrix}$$

where $I_{[N]}$ is an $N \times N$ identity matrix, the Walsh–Kaczmarz and Walsh–Hadamard spectral coefficients may be derived. The calculations are shown in Figs. 3 and 4, respectively.

To obtain back non-normalised Haar spectral coefficients, eqns. 12 and 13 and the generator matrix G will be used as follows: for $m = 1$

$$\mathbf{F}_{\mathbf{H}}^{\langle 1 \rangle} = \left(G^{\langle 1 \rangle} \diamond \mathbf{F}_{\mathbf{WP}}^{\langle 1 \rangle}\right) = \begin{bmatrix} F_H(0) \\ F_H(1) \end{bmatrix}$$
$$= [\phi] \diamond \begin{bmatrix} F_{WP}(0) \\ F_{WP}(1) \end{bmatrix} = \begin{bmatrix} 4 \\ 0 \end{bmatrix}$$

for $m = 2$

$$\mathbf{F}_{\mathbf{H}}^{\langle 2 \rangle} = \frac{1}{2}\left(G^{\langle 2 \rangle} \diamond \mathbf{F}_{\mathbf{WP}}^{\langle 2 \rangle}\right) = \begin{bmatrix} F_H(2) \\ F_H(3) \end{bmatrix}$$
$$= \frac{1}{2}\left(\begin{bmatrix} + \\ - \end{bmatrix} \diamond \begin{bmatrix} F_{WP}(2) \\ F_{WP}(3) \end{bmatrix}\right)$$
$$= \frac{1}{2}\begin{bmatrix} F_{WP}(2) + F_{WP}(3) \\ F_{WP}(2) - F_{WP}(3) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

and for $m = 3$

$$\mathbf{F}_{\mathbf{H}}^{\langle 3 \rangle} = \frac{1}{4}\left(G^{\langle 3 \rangle} \diamond \mathbf{F}_{\mathbf{WP}}^{\langle 3 \rangle}\right) = \begin{bmatrix} F_H(4) \\ F_H(5) \\ F_H(6) \\ F_H(7) \end{bmatrix}$$
$$= \frac{1}{4}\left(\begin{bmatrix} + & + & + \\ + & - & + \\ - & + & - \\ - & - & - \end{bmatrix} \diamond \begin{bmatrix} F_{WP}(4) \\ F_{WP}(5) \\ F_{WP}(6) \\ F_{WP}(7) \end{bmatrix}\right) = \begin{bmatrix} 1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

Therefore, $\mathbf{F_H} = [4, 0, 0, 0, 1, -1, -1, -1]^T$.

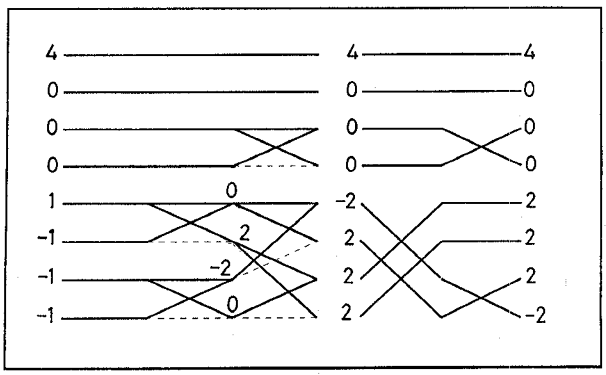

Fig.3 *Fast flow diagram of Haar–Walsh transform and bit permutation algorithm for f = [1, 0, 0, 1, 0, 1, 0, 1]^T*

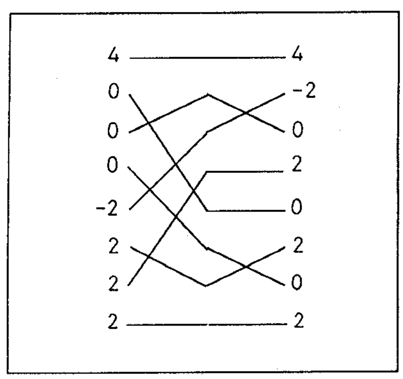

Fig.4 *Derivation of Walsh–Hadamard spectral coefficients from Walsh–Paley spectral coefficients by bit reversal algorithm*

7 Conclusions

A novel and unique linear transform, named the 'Haar–Walsh transform' has been introduced. A fast flow diagram for the implementation of the new transform in hardware has been shown. A novel generator matrix with new operations on matrices has been introduced. The matrix is used for efficient generation of HWT. Using the generator matrix, one is able to compute any single spectral coefficient without calculating the rest of the spectrum. Similarly to other Walsh-like transforms, the HWT can be generalised, i.e., it may be allowed to have negated variables for the basis switching functions constituting the transformation matrix. The ways to generalise the Walsh transform were shown in [23, 26], and can also be applied directly to the new transform. When the transform is generalised, it has 2^n different polarities and corresponding canonical polynomial expansions for the representation of an n-variable switching function [23, 26]. The HWT has an easy interpretation on the Karnaugh map similar to that for the Walsh transform [27]. Besides using the generator matrix to calculate single coefficients, it is also possible to relate such single coefficients directly to a reduced representation of switching functions such as cubes or different decision diagrams in a manner similar to earlier works relating such different representations with their Walsh-like spectra [10, 20, 27].

The fast HWT flow diagram has been developed to facilitate the calculation of the HWT. Though the flow graph can be efficiently implemented in software, its distinct properties suggest the possibility of developing a specialised in-place parallel processing VLSI hardware accelerator for its computation. Such a chip, designed for N-point fast HWT, can be used uniformly without the necessity of any reswitching for all lower dimensions of fast HWT. Moreover, only one hardware implementation is required since both forward and inverse HWT are exactly matching. The important properties and definitions of the fast HWT are listed in this paper. Since such a transform very efficiently converts Haar to Walsh spectral domains, it is no longer necessary to calculate the computationally expensive fast Walsh transform, as was clearly indicated by the results shown in Table 2. Hence all the presented results are not only very interesting theoretically but also are very important for the practical applications of Haar, Walsh and Walsh-like transforms in the designing of digital signal processing systems.

8 References

1 BEAUCHAMP, K.G.: 'Applications of Walsh and related functions' (Academic Press, London, 1984)

2 ELLIOTT, D.F., and RAO, K.R.: 'Fast transforms, algorithms, analyses, applications' (Academic Press, London, 1982)

3 HARMUTH, H.F.: 'Transmission of information by orthogonal functions' (Springer-Verlag, New York, 1972)

4 KULESZA, W.: 'Systems for spectral analysis of digital data' (WKL, Warsaw, 1984) (in Polish)

5 STOJIC, M.R., STANKOVIC, M.S., and STANKOVIC, R.S.: 'Discrete transforms and their application' (Nauka, Beograd, 1993) (in Serbian)

6 YAROSLAVSKY, L.P.: 'Digital picture processing – an introduction' (Springer-Verlag, New York, 1985)

7 ZALMANZON, L.A.: 'Fourier, Walsh and Haar transforms and their application in control, communication and other fields' (Nauka, Moscow, 1989) (in Russian)

8 GOLUBOV, B., EFIMOV, A., and SKVORTSOV, V.: 'Theory and applications of Walsh series and transforms' (Kluwer Academic, Boston, 1991)

9 KARPOVSKY, M.G.: 'Finite orthogonal series in the design of digital devices' (J. Wiley, New York, 1975)

10 HURST, S.L.: 'The logical processing of digital signals' (Crane-Russak, New York, 1978)

11 RAO, K.R., NARASIMHAN, M.A., and REVULIERI, K.: 'Image data processing by Hadamard–Haar transform', *IEEE Trans.*, 1975, **C-24**, pp. 888–896

12 RAO, K.R., JALALI, A., and YIP, P.: 'Rationalised Hadamard–Haar transform'. Proceedings of 11th Asilomar conference on *Circuits, systems and computers*, 1977, pp. 194–203

13 LI, Z.H., and ZHANG, Q.S.: 'Introduction to bridge functions', *IEEE Trans.*, 1983, **EC-25**, (4), pp. 459–464

14 ZHANG, Q.S.: 'A summary of bridge functions'. Proceedings of the 5th international workshop on *Spectral techniques*, 1994, Beijing, China, pp. 128–135

15 ZHANG, Q.S., and ZHANG, Y.G.: 'Theory and application of Bridge functions' (Defense Industry Publisher, Beijing, 1992) (in Mandarin)

16 SWICK, D.A.: 'Walsh function generation', *IEEE Trans.*, 1969, **IT-15**, pp. 167–168

17 KACZMARZ, S., and STEINHAUS, H.: 'Theory of orthogonal series' (Chelsea, New York, 1951) (in German)

18 FINO, B.J.: 'Relations between Haar and Walsh–Hadamard transform', *Proc. IEEE*, 1972, **60**, (5), pp. 647–648

19 STANKOVIC, R.S., and LUKOVIC, M.: 'Relations between some systems of orthogonal functions', *Automatika*, 1977, **1–2**, pp. 30–33 (in Serbian)

20 HURST, S.L., MILLER, D.M., and MUZIO, J.C.: 'Spectral techniques in digital logic' (Academic Press, London, 1985)

21 ZHANG, Q.S.: 'New methods of signal information transfer' (Aeronautics and Astronautics Publishers, Beijing, 1989) (in Mandarin)

22 WALLIS, W.D., STREET, A.P., and WALLIS, J.S.: 'Combinatorics: room squares, sum-free sets, Hadamard matrices', Lecture Notes in Mathematics, No. 292, (Springer-Verlag, Berlin, 1972)

23 FALKOWSKI, B.J.: 'Recursive relationships, fast transforms, generalisations and VLSI iterative architecture for Gray code ordered Walsh functions', *IEE Proc. Comput. Digit. Tech.*, 1995, **142**, (5), pp. 325–331

24 IRSHID, M.I.: 'A simple recursive definition for Walsh functions', *IEEE Trans.*, 1986, **EC-28**, (4), pp. 276–279

25 STANKOVIC, R.S.: 'On Haar and Walsh matrices', *Automatika*, 1977, **3–4**, pp. 102–107 (in Serbian)

26 FALKOWSKI, B.J.: 'Properties and ways of calculation of multipolarity generalised Walsh transforms', *IEEE Trans. Circuits Syst.–II: Analog Digital Signal Process.*, 1994, **41**, (6), pp. 380–391

27 FALKOWSKI, B.J., SCHAEFER, I., and PERKOWSKI, M.A.: 'Effective computer methods for the calculation of Radamacher-Walsh spectrum for completely and incompletely specified Boolean functions', *IEEE Trans.*, 1992, **CAD-11**, (10), pp. 1207–1226