Generalized Jacket Transform with Arbitrary Integer Entries

Shiang-Chih Hua and Jian-Jiun Ding
Graduate Institute of Communication Engineering, National Taiwan University, Taiwan
E-mail: f05942067@ntu.edu.tw, jjding@ntu.edu.tw

***Abstract*—In this paper, we further generalize the Jacket transform, which is a generalization of the Walsh transform. We find that, even if the entry is not a power of 2, with some constraint, the transform is still reversible and the inverse transform can be implemented using finite number of bits.**

I. Introduction

The integer transform is a discrete transform whose entries in both the forward and the inverse transform matrix are all integers or a sum of the power of 2:

$$\mathbf{A}=\begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,N} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N,1} & a_{N,2} & \cdots & a_{N,N} \end{bmatrix},\quad \mathbf{A}^{-1}=\begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,N} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ b_{N,1} & b_{N,2} & \cdots & b_{N,N} \end{bmatrix},\quad (1)$$

$$\mathbf{A}\mathbf{A}^{-1}=\mathbf{A}^{-1}\mathbf{A}=\mathbf{I}\,, \quad (2)$$

where $a_{m,n}$ are all integers and $b_{m,n}$ should have the form of

$$b_{m,n}=C_{m,n}/2^{k} \quad (3)$$

where $C_{m,n}$ is some integer. Since the entries of the forward transform are integers and those of the inverse transform have the form as in (3), they can be implemented precisely using finite number of bits. Thus, if the computation efficiency is an important issue, it is preferred to use the integer transform.

For example, the Walsh (Hadamard) transform is an integer transform. It is widely used in code division multiple access (CDMA). However, since its entries are restricted to ±1, in signal analysis, its performance is not as good as that of the discrete Fourier transform and the cosine transform (DCT). In [1, 2], Lee *et al.* proposed the Jacket transform, which is a generalization of the Walsh transform. Instead of setting entries to ±1, in the Jacket transform, the entries can be powers of 2. For example, the original 4-point Walsh transform is

$$\mathbf{W}_4=\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix},\quad \mathbf{W}_4^{-1}=\frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}. \quad (4)$$

By contrast, one of the 4-point Jacket transforms is

$$\mathbf{J}_4=\begin{bmatrix} 1 & x & x & 1 \\ 1 & x & -x & -1 \\ 1 & -x & -x & 1 \\ 1 & -x & x & -1 \end{bmatrix},\quad \mathbf{J}_4^{-1}=\frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ x^{-1} & x^{-1} & -x^{-1} & -x^{-1} \\ x^{-1} & -x^{-1} & -x^{-1} & x^{-1} \\ 1 & -1 & 1 & -1 \end{bmatrix} \quad (5)$$

where x is powers of 2:

$$x=2^{k}\,. \quad (6)$$

One can verify that $\mathbf{J}_4^{-1}\mathbf{J}_4=\mathbf{I}$. Although the entries are no longer ±1, since the multiplication of $\pm 2^k$ is easy to implement, the forward and inverse Jacket transform can still be implemented efficiently. The Jacket transform can be used for communication and signal expansion. Based on the similar concept, the Jacket Haar transform [3], the Walsh-Jacket transform with arbitrary length [4], and the Jacket-Haar transform [5] were proposed.

In this paper, we further generalize the Jacket transform. Instead of setting the entries to be a power of 2, we let the entry can be any other integers. We find that, if some constraints are satisfied, the generalized Jacket transform is still reversible and the entries of the inverse transform matrix will have the form as in (3), which can be implemented by finite number of bits. With the proposed method, it is possible to make the row of the transform matrix similar to the row of some poplar non-integer transform, such as the DCT.

II. 2-Point and 4-Point Generalized Jacket Transforms

We first review the way to construct the 2^p-point Walsh transforms. The 2-point Walsh transform and its inverse is

$$\mathbf{W}_2=\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},\quad \mathbf{W}_2^{-1}=\frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad (7)$$

The 2^L-point Walsh transform can be generated by the 2^{L-1}-point Walsh transform from

$$\mathbf{W}_{2^L}=\mathbf{Q}\left(\mathbf{W}_2\otimes\mathbf{W}_{2^{L-1}}\right),\quad \mathbf{W}_{2^L}^{-1}=\left(\mathbf{W}_2^{-1}\otimes\mathbf{W}_{2^{L-1}}^{-1}\right)\mathbf{Q}^{\mathrm{T}} \quad (8)$$

where ⊗ means the Kronecker product and **Q** is the permutation matrix that permutes the rows according to the number of zero crossings. From (8), we can say that the 2^L-point Walsh transform is generated based on the 2×2 Walsh transform matrix. Therefore, to generalized the definition of the Walsh transform, one can first generalize the 2×2 Walsh transform matrix.

First, we note that if

$$\mathbf{G}_2=\begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (9)$$

where $ad-bc=-2^k$, (10)

then the inverse of $\mathbf{G}_2$ is

$$\mathbf{G}_2^{-1}=\begin{bmatrix} -2^{-k}d & 2^{-k}b \\ 2^{-k}c & -2^{-k}a \end{bmatrix}. \quad (11)$$

That is, even if the entries a, b, c, d are not powers of 2, if they satisfies the constraint in (10), then the entries of the inverse transform matrix have the form as in (3) and can be implemented by finite number of bits. Therefore, one can define the 2-point generalized Jacket transform as in (9) and the entries a, b, c, d should satisfy the following two constraints:

(i) $ad-bc=-2^k$, (ii) $a>0$, $b>0$, $c>0$, $d<0$. (12)

The 2nd constraint is to match the entry sign of the original 2-point Walsh transform.

Since the original 4-point Walsh transform is generated from the 2-point Walsh transform as follows:

$$\mathbf{W}_4=\mathbf{Q}\left(\mathbf{W}_2\otimes\mathbf{W}_2\right)$$

to define the 4-point generalized Jacket transform, one can replace two $\mathbf{W}_2$ in (12) by the transform matrices of the 2-point generalized Jacket transforms.

• **4-point generalized Jacket transform matrix**

$$\mathbf{G_4} = \mathbf{Q}\left(\mathbf{G_2} \otimes \mathbf{H_2}\right)\mathbf{P} \tag{13}$$

where $$\mathbf{G_2} = \begin{bmatrix} a_1 & b_1 \\ c_1 & d_1 \end{bmatrix}, \quad \mathbf{H_2} = \begin{bmatrix} a_2 & b_2 \\ c_2 & d_2 \end{bmatrix}, \tag{14}$$

$$a_1d_1 - b_1c_1 = -2^{k_1}, \quad a_1 > 0, \quad b_1 > 0, \quad c_1 > 0, \quad d_1 < 0,$$

$$a_2d_2 - b_2c_2 = -2^{k_2}, \quad a_2 > 0, \quad b_2 > 0, \quad c_2 > 0, \quad d_2 < 0, \tag{15}$$

and **Q** and **P** are permutation matrices. The inverse of (13) is

$$\mathbf{G_4^{-1}} = \mathbf{P^T}\left(\mathbf{G_2^{-1}} \otimes \mathbf{H_2^{-1}}\right)\mathbf{Q^T} \tag{16}$$

where $$\mathbf{G_2^{-1}} = \begin{bmatrix} -2^{-k_1}d_1 & 2^{-k_1}b_1 \\ 2^{-k_1}c_1 & -2^{-k_1}a_1 \end{bmatrix}, \quad \mathbf{H_2^{-1}} = \begin{bmatrix} -2^{-k_2}d_2 & 2^{-k_2}b_2 \\ 2^{-k_2}c_2 & -2^{-k_2}a_2 \end{bmatrix}. \tag{17}$$

For example, we can choose

$$\mathbf{G_2} = \begin{bmatrix} 1 & 1 \\ 3 & -1 \end{bmatrix}, \quad \mathbf{H_2} = \begin{bmatrix} 2 & 3 \\ 2 & -1 \end{bmatrix}. \tag{18}$$

Note that $a_1d_1 - b_1c_1 = -4$, $a_2d_2 - b_2c_2 = -8$, and all the constraints in (15) are satisfied. Then, we choose

$P(0,0) = P(1,1) = P(3,2) = P(2,3) = 1$,
$Q(0,0) = Q(2,1) = Q(1,2) = Q(3,3) = 1$,
$P(n, j) = Q(n, j) = 0$ otherwise. (19)

Then, the forward/inverse 4-pt generalized Jacket transforms are

$$\mathbf{G_4} = \begin{bmatrix} 2 & 3 & 3 & 2 \\ 6 & 9 & -3 & -2 \\ 2 & -1 & -1 & 2 \\ 6 & -3 & 1 & -2 \end{bmatrix}, \quad \mathbf{G_4^{-1}} = \frac{1}{32}\begin{bmatrix} 1 & 1 & 3 & 3 \\ 2 & 2 & -2 & -2 \\ 6 & -2 & -6 & 2 \\ 3 & -1 & 9 & -3 \end{bmatrix}. \tag{20}$$

One can verify that $\mathbf{G_4}\mathbf{G_4}^{-1} = \mathbf{I}$ is satisfied. Note that, different from the Walsh transform in (4) and the Jacket transform in (5), 7 of the entries of $\mathbf{G_2}$ is not a power of 2. However, since the entries of the inverse transform matrix $\mathbf{G_4}^{-1}$ has the form as in (3), the generalized Jacket transform is still an integer transform and can be implemented by finite number of points. The original 4-point Jacket transform in (5) can be viewed as a special case of the 4-point generalized Jacket transform in (13) where

$$\mathbf{G_2} = \begin{bmatrix} 1 & x \\ 1 & -x \end{bmatrix}, \; x = 2^k, \; \mathbf{H_2} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \; \mathbf{P} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{21}$$

and **Q** = **I**. Note that the determinants of both the two 2×2 matrices in (21) are powers of 2.

III. 2^L-Point Generalized Jacket Transform

We can also apply the way similar to that in (13) to define the 8-point, 16-point, and 2^L-point generalized Jacket transform.

• **2^L-point generalized Jacket transform matrix**

$$\mathbf{G}_{2^L} = \mathbf{Q}\big(\underbrace{\mathbf{G_{2,1}} \otimes \mathbf{G_{2,2}} \otimes \cdots \otimes \mathbf{G_{2,L}}}_{L \text{ terms}}\big)\mathbf{P} \tag{22}$$

where **Q** and **P** are permutation matrices and $\mathbf{G_{2,1}}$, $\mathbf{G_{2,2}}$, ……, $\mathbf{G}_{2,L}$ are all 2×2 matrices that satisfy (9) and (12):

$$\mathbf{G}_{2,m} = \begin{bmatrix} a_m & b_m \\ c_m & d_m \end{bmatrix}, \tag{23}$$

where a_m, b_m, c_m, d_m can be any integers satisfying:

$$a_md_m - b_mc_m = -2^{k_m}, \quad a_m > 0, \quad b_m > 0, \quad c_m > 0, \quad d_m < 0. \tag{24}$$

The inverse 2^L-point generalized Jacket transform matrix is

$$\mathbf{G}_{2^L}^{-1} = \mathbf{P^T}\left(\mathbf{G_{2,1}^{-1}} \otimes \mathbf{G_{2,2}^{-1}} \otimes \cdots \otimes \mathbf{G_{2,L}^{-1}}\right)\mathbf{Q^T} \tag{25}$$

where $$\mathbf{G}_{2,m}^{-1} = \begin{bmatrix} -2^{-k_m}d_m & 2^{-k_m}b_m \\ 2^{-k_m}c_m & -2^{-k_m}a_m \end{bmatrix}. \tag{26}$$

One can verify that $\mathbf{G}_{2^L}\mathbf{G}_{2^L}^{-1} = \mathbf{I}$ if (24) is satisfied.

Since for the generalized Jacket transform the entry are more flexible to choose, it is possible to use the generalized Jacket transform to approximate some well-known discrete operation, such as the DCT. The 8-point DCT is

$$C[n, j] = k_n \cos\left(\pi n(j + 1/2)/8\right) \quad n, j = 0, 1, \ldots, 7,$$

$$k_0 = \sqrt{1/8}, \qquad k_n = 1/2 \quad \text{for } n \neq 0. \tag{27}$$

If the 8-point generalized Jacket transform and its parameters are:

$$\mathbf{G_8} = \mathbf{Q}\left(\mathbf{G_{2,1}} \otimes \mathbf{G_{2,2}} \otimes \mathbf{G_{2,3}}\right)\mathbf{P} \tag{28}$$

where $$\mathbf{G_{2,1}} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \mathbf{G_{2,2}} = \mathbf{G_{2,3}} = \begin{bmatrix} 1 & 1 \\ 3 & -5 \end{bmatrix}, \tag{29}$$

$P(0,0)=P(5,1)=P(6,2)=P(3,3)=P(7,4)=P(2,5)=P(1,6)=P(4,7)= 1$, $Q(0,0)=Q(7,1)=Q(2,2)=Q(5,3)=Q(3,4)=Q(4,5)=Q(1,6)=Q(8,7)$ $=1$, $P(n, j) = Q(n, j) = 0$ otherwise. (30)

Then, the 8-point generalized Jacket transform we obtain is

$$\mathbf{G_8} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 25 & 15 & 15 & 9 & -9 & -15 & -15 & -25 \\ 5 & 5 & -3 & -3 & -3 & -3 & 5 & 5 \\ 5 & 3 & -5 & -3 & 3 & 5 & -3 & -5 \\ 25 & -15 & -15 & 9 & 9 & -15 & -15 & 25 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 5 & -3 & 5 & -3 & -3 & 5 & -2 & 5 \\ 5 & -5 & 3 & -3 & 3 & -3 & 5 & -5 \end{bmatrix}. \tag{31}$$

Compared to the Walsh and Jacket transform matrices, after normalization, the waveform of the matrix in (31) is more similar to that of the 8-point DCT. After normalization, the sum of square differences of the first 4 rows (low frequency part) between the 8-point DCT and (31) is 0.5096. By contrast, that between the 8-point DCT and the 8-point Walsh or Jacket transform is 0.8036.

IV. Conclusion

A further generalization version of the Jacket transform is proposed. Instead of setting the entries to ±1 or $\pm 2^k$, the entries of the generalized Jacket transform can be any integer. The only constraints are that the 2×2 matrices used for constructing the transform should satisfy (24). As the original Walsh or Jacket transform, it can be implemented efficiently using finite number of bits. Since the entries can be set more flexibly, the derived generalized Jacket transform can well approximate some well-known discrete operations, such as the DCT.

References

[1] M. H. Lee, “A new reverse jacket transform and its fast algorithm,” I*EEE Trans. Circuits and Syst. II: Analog and Digit. Signal Process.*, vol. 47, issue 1, pp. 39-47, 2000.

[2] M. H. Lee and K. Finlayson. "A simple element inverse Jacket transform coding," *IEEE Signal Process. Lett.*, vol. 14, issue 5 pp. 325-328, 2007.

[3] J. J. Ding, S. C. Pei, and P. H. Wu, “Jacket Haar transform,” ISCAS, Rio de Janeiro, Brazil, pp. 1520-1523, May 2011.

[4] J. J. Ding, S. C. Pei, and P. H. Wu, “Arbitrary-length Walsh-Jacket transforms,” *APSIPA Annual Summit and Conference*, Xi'an, China, pp. 1-10, Oct. 2011.

[5] G. Liu, D. Luo, G. Lv, Y. Guo, and M. Lee, “Arbitrary-length Jacket-Haar transforms,” in *Int. Conf. Algorithms and Architectures for Parallel Processing*, pp. 330-343, Nov. 2015.