Journal of Mathematical Imaging and Vision 16: 31–39, 2002
© 2002 Kluwer Academic Publishers. *Manufactured in The Netherlands.*

Fast Reverse Jacket Transform As an Alternative Representation of the *N*-Point Fast Fourier Transform*

SEUNG-RAE LEE
Institute of New Media and Communications, Seoul National University, Seoul 151-742, Korea
srlee@acoustics.snu.ac.kr

JUNE-HO YI
School of Electrical and Computer Engineering, Sungkyunkwan University, Suwon 440-746, Korea
jhyi@yurim.skku.ac.kr

Abstract. The Reverse Jacket matrix (RJM) is a generalized form of the Hadamard matrix. Thus RJM is closely related to the matrix for fast Fourier transform (FFT). It also has a very interesting structure, i.e. its inverse can be easily obtained and has the reversal form of the original matrix. In this paper, we have shown that a transform based on the RJM offers a simple structure of N-point FFT in terms of the decomposition of the corresponding matrix and that it computes very fast the center weighted Hadamard transform.

Keywords: Hadamard matrix, Reverse Jacket matrix, fast Fourier transform, fast Reverse Jacket transform

1. Introduction

A Hadamard matrix of order $N \in \mathbb{N}$ is an $N \times N$ matrix of ones and minus ones such that any two distinct rows are orthogonal. The Hadamard transform, y, of the N-dimensional vector x is defined with a $N \times N$ Hadamard matrix $\mathbf{H}_N$ as

$$y = \mathbf{H}_N x.$$

For the purpose of developing a fast algorithm for the Fourier transform, the Hadamard transform has often been employed [1, 14]. The Fourier transform is one of the most fundamental transforms used in signal processing and analysis. The Hadamard transform (HT) is an orthogonal transform for representing signals and images, especially for data compression. Hadamard matrices and their transforms have more recently been applied to information theory, signal and image processing and error-correction coding because of their efficiency due to orthogonality [1, 10, 14].

The Reverse Jacket matrix was first defined in [8]. It is quite useful to represent a modified Hadamard matrix where all or some elements of the Hadamard matrix need to be other numbers rather than ± 1. For the same purpose, the center-weighted Hadamard transform (CWHT) was proposed in [6], however, its structure is neither as general nor as computationally efficient as the RJM based transform.

The nice property of RJMs is that the inverse of the RJM can be obtained very easily and has a special structure [8]. In [9] the RJM having a reverse geometric structure was generalized. This property enables 1-D fast Reverse Jacket transform (FRJT) to provide quite a simple factorization of the matrix products in terms of the direct sum, Kronecker product and usual matrix multiplications. The number of arithmetic operations required in the FRJT is the same as in the regular FFT, which is $O(N \log_2 N)$. However, FRJT requires less arithmetic operations than CWHT [6]. The amount of computation required for CWHT is $KN + \frac{N}{2}$ real

*This research was supported in part by grant no. 1999-2-515-001-5 from the Basic Research Program of the KOSEF and by the Brain Korea 21 program of the Minister of Education, Korea.

additions and $\frac{3N}{2}$ real multiplications for $N = 2^K$. K and N denote the multiplicity of 2 and the dimension of the input vector, respectively. In addition, FRJT is applicable to complex-valued matrices while only real matrix entries and symmetric basic matrices were considered in CWHT.

In this paper, we present 1-D fast Reverse Jacket transform (FRJT) based RJMs and analyze the relationship between FRJT and FFT. We also show that FRJT is computationally as efficient as FFT. The main advantage of using FRJT is three folds. First, when the central parts of data sequences or the middle ranges of frequency components need to be stressed, FRJT that generalizes CWHT is more appropriate to use than the HT. Second, this FRJT is remarkably faster than the algorithms reported in [6, 7]. Third, FRJT is another way of representing FFT. The N-point FFT can be expressed in terms of matrix decomposition by using 4×4 FRJT.

The paper is structured as follows. Section 2 describes the definition of RJMs and some preliminaries. Section 3 gives some results about fast 1-D Reverse Jacket transform in which each element of the basic matrix can be an arbitrary number as opposed to HT and CWHT. In Section 4 we illustrate how a 4-point FFT can be achieved by 1-D FRJT. The decomposition of the N-point FFT matrix is derived from that of the 4×4 RJM. We also show that an application of FRJT to the N-point fast Fourier transform.

2. The Reverse Jacket Matrix and Preliminaries

For a given 2×2 basic matrix $\mathbf{R}_2 \in I\!R^{2\times 2}$ we construct an invertible 4×4 matrix as a generalizing Hadamard matrix, called a Reverse Jacket matrix (RJM). The RJM has a reverse geometric structure. As our two-sided jacket is compatible inside and outside, at least two positions of a Reverse Jacket matrix are replaced by their inverses. These entries change their positions and move, for example, from the inside of the middle circle to the outside or from the outside to the inside without loss of signs in the cases of symmetric matrices, or with the interchange of signs of certain groups in the cases of asymmetric matrices. This is the reason why we call them Reverse Jacket matrices [8]. Given a 2×2 basic matrix $\mathbf{R}_2 \in I\!R^{2\times 2}$ we will construct an invertible 4×4 matrix. Let $\mathbf{I}_{2^j}$ and $\mathbf{0}_{2^j}$, $j \in I\!N \cup 0$ be the $(2^j \times 2^j)$ unit matrix and zero matrix, respectively.

Definition 2.1 ([9]). Assume that a, b, c, d are nonzero complex numbers. The resultant Reverse Jacket matrix (RJM) is defined by

$$\mathbf{R}_{2^{k+1}} = \left[\begin{array}{ccc} \mathbf{R}_{2^k} & \vdots & \mathbf{Z}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{S}_{2^k} \\ \cdots\cdots\cdots & \cdots & \cdots\cdots\cdots \\ \mathbf{S}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{Z}_{2^k} & \vdots & -\mathbf{J}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{J}_{2^k} \end{array}\right] \tag{2.1}$$

with initial conditions

$$\mathbf{R}_2 := \begin{bmatrix} a & b \\ c & -d \end{bmatrix}, \text{ and } \mathbf{Z}_{2^k} := \begin{bmatrix} I_{2^{(k-1)}} & \mathbf{0}_{2^{(k-1)}} \\ \mathbf{0}_{2^{(k-1)}} & -I_{2^{(k-1)}} \end{bmatrix},$$

$$\mathbf{S}_{2^k} := \begin{bmatrix} \mathbf{0}_{2^{(k-1)}} & \mathbf{I}_{2^{(k-1)}} \\ \mathbf{I}_{2^{(k-1)}} & \mathbf{0}_{2^{(k-1)}} \end{bmatrix}, \text{ and}$$

$$\mathbf{J}_{2^k} := \begin{bmatrix} \mathbf{0}_{2^{(k-1)}} & \mathbf{I}_{2^{(k-1)}} \\ -\mathbf{I}_{2^{(k-1)}} & \mathbf{0}_{2^{(k-1)}} \end{bmatrix} \quad \text{for } k \geq 1. \tag{2.2}$$

Theorem 2.2 ([9]). *Assume that* $\mathbf{R}_2$ *is nonsingular. If*

$$\mathbf{R}_{2^{k+1}} = \begin{bmatrix} \mathbf{R}_{2^k} & \mathbf{Z}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{S}_{2^k} \\ \mathbf{S}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{Z}_{2^k} & -\mathbf{J}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{J}_{2^k} \end{bmatrix},$$

then $\mathbf{R}_{2^{k+1}}$ *is nonsingular with*

$$\mathbf{R}_{2^{k+1}}^{-1} = \begin{bmatrix} \check{\mathbf{R}}_{2^k}^{-1} & \mathbf{Z}_{2^k}\,\check{\mathbf{R}}_{2^k}^{-1}\,\mathbf{S}_{2^k} \\ \mathbf{S}_{2^k}\,\check{\mathbf{R}}_{2^k}^{-1}\,\mathbf{Z}_{2^k} & -\mathbf{J}_{2^k}\,\check{\mathbf{R}}_{2^k}^{-1}\,\mathbf{J}_{2^k} \end{bmatrix} \tag{2.3}$$

where $\check{\mathbf{R}}_{2^k} = \mathbf{R}_{2^k} + \mathbf{Z}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{S}_{2^k}\,\mathbf{J}_{2^k}\,\mathbf{R}_{2^k}^{-1}\,\mathbf{J}_{2^k}\,\mathbf{S}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{Z}_{2^k}$.

Proof: See appendix. □

Remark 2.3. It is easily verified that

$$\check{\mathbf{R}}_2^{-1} = \frac{1}{4} \cdot \begin{bmatrix} \frac{1}{a} & \frac{1}{c} \\ \frac{1}{b} & -\frac{1}{d} \end{bmatrix}.$$

Hence, we can construct the inverse of $\mathbf{R}_4$ very easily and the inverses of higher order RJMs can be obtained in the same way.

Theorem 2.4 ([9]). *Assume that* $\mathbf{R}_2$ *is nonsingular. If*

$$\mathbf{X}_{2^{k+1}} = \begin{bmatrix} \mathbf{R}_{2^k} & \mathbf{Z}_{2^k}\,\mathbf{R}_{2^k} \\ \mathbf{R}_{2^k}\,\mathbf{Z}_{2^k} & \mathbf{Z}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{Z}_{2^k} \end{bmatrix},$$

then $\mathbf{X}_{2^{k+1}}$ *is nonsingular with*

$$\mathbf{X}_{2^{k+1}}^{-1} = \begin{bmatrix} \hat{\mathbf{R}}_{2^k}^{-1} & \mathbf{Z}_{2^k}\,\hat{\mathbf{R}}_{2^k}^{-1} \\ \hat{\mathbf{R}}_{2^k}^{-1}\,\mathbf{Z}_{2^k} & \mathbf{Z}_{2^k}\,\hat{\mathbf{R}}_{2^k}^{-1}\,\mathbf{Z}_{2^k} \end{bmatrix} \tag{2.4}$$

where $\hat{\mathbf{R}}_{2^k} = \mathbf{R}_{2^k} - \mathbf{Z}_{2^k}\mathbf{R}_{2^k}\mathbf{Z}_{2^k}\mathbf{R}_{2^k}^{-1}\mathbf{Z}_{2^k}\mathbf{R}_{2^k}\mathbf{Z}_{2^k}$ *for* $k \geq 1$.

Proof: Analogous to the proof of Theorem 2.2. □

Corollary 2.5 ([9]). *An orthogonal matrix* $\mathbf{U}_{2^{k+1}}$ *exists for* $k \geq 1$ *such that*

$$\mathbf{R}_{2^{k+1}} = \mathbf{U}_{2^{k+1}}\mathbf{X}_{2^{k+1}}\mathbf{U}_{2^{k+1}}.$$

Proof: See appendix. □

Example 2.6. Let $\mathbf{R}_2$ be a basic matrix for the RJM defined by $\mathbf{R}_2 := \left[\begin{smallmatrix} 4 & 1 \\ -1 & -2 \end{smallmatrix}\right]$. Then it holds

$$\mathbf{R}_4 = \begin{bmatrix} 4 & 1 & 1 & 4 \\ -1 & -2 & 2 & 1 \\ -1 & 2 & -2 & 1 \\ 4 & -1 & -1 & 4 \end{bmatrix},$$

$$\mathbf{R}_4^{-1} = \frac{1}{16}\begin{bmatrix} 1 & -4 & -4 & 1 \\ 4 & -2 & 2 & -4 \\ 4 & 2 & -2 & -4 \\ 1 & 4 & 4 & 1 \end{bmatrix}.$$

Here the matrix elements ± 1 of $\mathbf{R}_4$ belongs to the middle circle, 4 to the corner, and ± 2 to the center region, respectively. Note that the middle circle and the corner regions of the Reverse Jacket matrix $\mathbf{R}_4$ are interchanged when $\mathbf{R}_4^{-1}$ is computed.

3. 1-D Fast Reverse Jacket Transform (FRJT)

Before presenting FRJT, let us give the definition of a direct sum of matrices and the Kronecker product.

Definition 3.1 ([13]). Let $V_1, \ldots, V_n$ be vector space over the same field $\mathcal{F}$. The (external) direct sum of $V_1, \ldots, V_n$, denoted by $V = V_1 \otimes \cdots \otimes V_n$ is the vector space V whose elements are ordered n-tuples

$$V = \{(v_1, \ldots, v_n) \mid v_i \in V_i,\ i = 1, \ldots, n\}$$

with componentwise operations

$$(u_1, \ldots, u_n) + (v_1, \ldots, v_n) = (u_1 + v_1, \ldots, u_n + v_n)$$

and

$$r(v_1, \ldots, v_n) = (rv_1, \ldots, rv_n), \quad r \in \mathbb{R}.$$

The block diagonal matrix

$$\mathbf{A} = \begin{bmatrix} \mathbf{A_1} & 0 \\ 0 & \mathbf{A_2} \end{bmatrix}$$

is said to be a direct sum of the matrices $\mathbf{A_1}$ and $\mathbf{A_2}$ and is written $\mathbf{A} = \mathbf{A_1} \oplus \mathbf{A_2}$ where $\mathbf{A_1} \in \mathcal{F}^{k\times k}$ and $\mathbf{A_1} \in \mathcal{F}^{(n-k)\times(n-k)}$ for $k,\ n \in \mathbb{N}$ (see [5]).

Definition 3.2 ([4]). The Kronecker product of $\mathbf{A} = [a_{ij}] \in \mathcal{F}^{m\times n}$ and $\mathbf{B} = [b_{ij}] \in \mathcal{F}^{p\times q}$ is denoted by $\mathbf{A} \otimes \mathbf{B}$ and defined to be the block matrix

$$\mathbf{A} \otimes \mathbf{B} := \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix} \in \mathcal{F}^{mp\times nq}.$$

Note that $\mathbf{A} \otimes \mathbf{B} \neq \mathbf{B} \otimes \mathbf{A}$ in general.

Now we present a fast algorithm for RJT. These algorithms are obtained by decomposition of the matrix using the permutation matrix, direct sum and Kronecker product. Only real entries of the basic matrix were considered in these transforms. We use two permutation matrices defined as

$$\mathbf{P}_{2^k} := \mathbf{P}_4 \otimes \mathbf{I}_{2^{k-2}}, \quad k \geq 2,$$

where $\mathbf{I}_{2^{k-2}}$ is the $k-2$-dimensional identity matrix and

$$\mathbf{P}_4 := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

The column matrix $\mathbf{Q}_{2^k},\ k \geq 2$ is defined as

$$\mathbf{Q}_{2^k} := [\mathbf{I}_{2^{k-1}} \oplus (\mathbf{S}_2 \otimes \mathbf{I}_{2^{k-2}})]$$

where $\mathbf{S}_2 := \left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$. Using the unitary properties of the permutation matrix, we decompose the RJM into a new form:

$$\mathbf{R}_{2^k} := \mathbf{P}_{2^k}^T\,\tilde{\mathbf{R}}_{2^k}\,\mathbf{Q}_{2^k}^T. \tag{3.5}$$

The following matrix identity is easily verified

$$\tilde{\mathbf{R}}_{2^k} = \begin{bmatrix} \mathbf{A} & \mathbf{A} \\ \mathbf{B} & -\mathbf{B} \end{bmatrix} = (\mathbf{A} \oplus \mathbf{B})\,(\mathbf{H}_2 \otimes \mathbf{I}_{2^{k-1}}), \quad (3.6)$$

where

$$\mathbf{A} = \begin{bmatrix} a & b \\ a & -b \end{bmatrix} \otimes \mathbf{H}_2^{k-2} =: \mathbf{V}_A \otimes \mathbf{H}_2^{k-2},$$
$$\mathbf{V}_A = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} =: \mathbf{H}_2\,\mathbf{U}_A, \quad (3.7)$$

and

$$\mathbf{B} = \begin{bmatrix} c & d \\ c & -d \end{bmatrix} \otimes \mathbf{H}_2^{k-2} =: \mathbf{V}_B \otimes \mathbf{H}_2^{k-2},$$
$$\mathbf{V}_B = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} c & 0 \\ 0 & d \end{bmatrix} =: \mathbf{H}_2\,\mathbf{U}_B. \quad (3.8)$$

Decomposing the above matrices gives

$$\begin{aligned} A &= \mathbf{V}_A \otimes \mathbf{H}_{2^{k-2}} = \mathbf{H}_2\,\mathbf{U}_A \otimes \mathbf{H}_{2^{k-2}}\,\mathbf{I}_{2^{k-2}} \\ &= \mathbf{H}_{2^{k-1}}(\mathbf{U}_A \otimes \mathbf{I}_{2^{k-2}}), \quad (3.9) \\ \mathbf{B} &= \mathbf{V}_B \otimes \mathbf{H}_{2^{k-2}} = \mathbf{H}_2\,\mathbf{U}_B \otimes \mathbf{H}_{2^{k-2}}\,\mathbf{I}_{2^{k-2}} \\ &= \mathbf{H}_{2^{k-1}}(\mathbf{U}_B \otimes \mathbf{I}_{2^{k-2}}). \quad (3.10) \end{aligned}$$

Hence, we have

$$\begin{aligned} \tilde{\mathbf{R}}_{2^k} &:= (\mathbf{A} \oplus \mathbf{B})\,(\mathbf{H}_2 \otimes \mathbf{I}_{2^{k-1}}) \\ &= [\{\mathbf{H}_{2^{k-1}}(\mathbf{U}_A \otimes \mathbf{I}_{2^{k-2}})\} \oplus \{\mathbf{H}_{2^{k-1}}(\mathbf{U}_B \otimes \mathbf{I}_{2^{k-2}})\}] \\ &\quad \times (\mathbf{H}_2 \otimes \mathbf{I}_{2^{k-1}}) \\ &= \underbrace{\underbrace{(\mathbf{H}_{2^{k-1}} \oplus \mathbf{H}_{2^{k-1}})}_{\text{third stage}}}_{N \log_2 \frac{N}{2} \text{ additions}} \underbrace{\underbrace{(\,(\mathbf{U}_A \otimes \mathbf{I}_{2^{k-2}}) \oplus (\mathbf{U}_B \otimes \mathbf{I}_{2^{k-2}})\,)}_{\text{second stage}}}_{N \text{ multiplications}} \\ &\quad \times \underbrace{\underbrace{(\mathbf{H}_2 \otimes \mathbf{I}_{2^{k-1}})}_{\text{first stage}}}_{N \text{ additions}}. \quad (3.11) \end{aligned}$$

Then, the Reverse Jacket matrix is nicely decomposed as follows:

$$\begin{aligned} \mathbf{R}_{2^k} &= \mathbf{P}_{2^k}^T\,\tilde{\mathbf{R}}_{2^k}\,\mathbf{Q}_{2^k}^T \\ &= \mathbf{P}_{2^k}^T\,(\mathbf{H}_{2^{k-1}} \oplus \mathbf{H}_{2^{k-1}})((\mathbf{U}_A \otimes \mathbf{I}_{2^{k-2}}) \oplus (\mathbf{U}_B \otimes \mathbf{I}_{2^{k-2}})) \\ &\quad \times (\mathbf{H}_2 \otimes \mathbf{I}_{2^{k-1}})\,\mathbf{Q}_{2^k}^T. \quad (3.12) \end{aligned}$$

The Reverse Jacket transform, y, of the N-dimensional vector x is defined with a $N \times N$ Reverse Jacket matrix $\mathbf{R}_N$ as

$$y = \mathbf{R}_N\,x. \quad (3.13)$$

As can be seen immediately in Eq. (3.12), this transform provides quite a simple factorization of the matrix products in terms of the direct sum, Kronecker product and usual matrix multiplications. We now check the computation at each stage of the FRJT algorithm. We have safely ignored the computational cost of data permutation in the analysis of computational complexity of the algorithm because its cost is much less compared to that of multiplication and addition. We first consider the operation at the third stage, which needs $N \log_2 \frac{N}{2}$ additions for $N = 2^k,\ k \geq 2$. The number of operations at the second stage is N multiplications, because we compute it from a diagonal matrix with $\frac{N}{4}$ entries of $a,\ b,\ c\ , d$, respectively. At the first stage, we need N additions. Hence, this algorithm requires

- $N(1 + \log_2 \frac{N}{2}) = N \log_2 N$ additions and N multiplications for $a, b, c, d \neq 1$
- $N \log_2 N$ real additions and $\frac{N}{4}$ real multiplications for $a = b = c = 1$ and $d = w \in I\!R$ [2]
- $N \log_2 N$ real additions and $\frac{N}{4}$ complex multiplications for $a = b = c = 1$ and $d = j \in \mathbb{C}$

Note that when $a = b = c = 1$ and $d = 2$, FRJT is equivalent to the center-weighted Hadamard transform reported in [2, 6]. The amount of arithmetic operations required for center-weighted Hadamard transform is $(KN + \frac{N}{2})$ real additions and $\frac{3N}{2}$ real multiplications for $N = 2^K$. Clearly 1-D FRJT is a faster algorithm than CWHT because, in addition, $N \log_2 N < (KN + \frac{N}{2})$ and in multiplication, $\frac{N}{4} < \frac{3N}{2}$. Notice that the number of arithmetic operations required in the FRJT is the same as in the regular FFT, both having the order of $N \log_2 N$.

4. Application to N-Point Fast Fourier Transform (FFT)

The N-point fast Fourier transform of a set of data, say, $x(0), x(1), \ldots, x(N-1)$ is given by the transform coefficients $X(0), X(1), \ldots, X(N-1)$ by the relation

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ \vdots \\ X(N-1) \end{bmatrix} = \begin{bmatrix} w_N^0 & w_N^0 & w_N^0 & \cdots & w_N^0 \\ w_N^0 & w_N^1 & w_N^2 & \cdots & w_N^{N-1} \\ w_N^0 & w_N^2 & w_N^4 & \cdots & w_N^{2(N-1)} \\ w_N^0 & w_N^3 & w_N^6 & \cdots & w_N^{3(N-1)} \\ \vdots & & \vdots & & \\ w_N^0 & w_N^{N-1} & w_N^{2(N-1)} & \cdots & w_N^{(N-1)(N-1)} \end{bmatrix} \cdot \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ \vdots \\ x(N-1) \end{bmatrix} \tag{4.14}$$

where $x(n)$ is the signal sequence and

$$w_N = e^{-j\frac{2\pi}{N}} = \cos(2\pi/N) - j \cdot \sin(2\pi/N).$$

The parameter w_N is an nth root of unity because $w_N^N = 1$. Equation (4.14) can be written as

$$X = \mathbf{W}_N x \tag{4.15}$$

where $X = [X(0), \ldots, X(N-1)]^T$ and $x = [x(0), \ldots, x(N-1)]^T$. The inverse transform is

$$x = \frac{1}{N}\mathbf{W}_N^* X \tag{4.16}$$

where $*$ denotes conjugation. $\mathbf{W}_N$ is said to be an FFT matrix.

In the following, we illustrate how a 4-point FFT can be achieved by 1-D FRJT.

Example 4.1. Let

$$\mathbf{W}_4 := \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \tag{4.17}$$

be the well-known 4-point FFT matrix. Let

$$\mathbf{R}_2 := \begin{bmatrix} 1 & 1 \\ 1 & -j \end{bmatrix} \quad \text{and} \quad \mathbf{Q}_4 := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

be the basic matrix and the permutation matrix for an RJM, respectively. We then obtain

$$\mathbf{R}_4 = \mathbf{Q}_4 \cdot \mathbf{W}_4 \cdot \mathbf{Q}_4, \quad \mathbf{W}_4 = \mathbf{Q}_4 \cdot \mathbf{R}_4 \cdot \mathbf{Q}_4. \tag{4.18}$$

Using this algorithm (3.12), we obtain $\mathbf{T}_4 := \mathbf{Q}_4 \cdot \mathbf{P}_4^T$ and $\mathbf{Q}_4^T \cdot \mathbf{Q}_4 = (\mathbf{Q}_4)^2 = \mathbf{I}_4$. Therefore

$$\begin{aligned} \mathbf{W}_4 &= \mathbf{Q}_4 \cdot \mathbf{P}_4^T\,(\mathbf{H}_2 \oplus \mathbf{H}_2)(\,(\mathbf{U}_A) \oplus (\mathbf{U}_B)\,)(\mathbf{H}_2 \otimes \mathbf{I}_2) \\ &\quad \times \mathbf{Q}_4^T \cdot \mathbf{Q}_4 \qquad (4.19) \\ &= \mathbf{T}_4\,(\mathbf{H}_2 \oplus \mathbf{H}_2)(\,(\mathbf{U}_A) \oplus (\mathbf{U}_B)\,)(\mathbf{H}_2 \otimes \mathbf{I}_2) \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \\ &\quad \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & j \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}. \qquad (4.20) \end{aligned}$$

The amount of computation needed at each stage of this example is 8 real additions and 1 complex multiplication in the case of $a = b = c = 1$ and $d = j \in \mathbb{C}$.

In the N = 8 case, $w_4^1 = w_8^2 = -j$, $w_4^2 = w_8^4 = -1$ and

$$\mathbf{W}_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & w_8^1 & w_8^2 & w_8^3 & w_8^4 & w_8^5 & w_8^6 & w_8^7 \\ 1 & w_8^2 & w_8^4 & w_8^6 & 1 & w_8^2 & w_8^4 & w_8^6 \\ 1 & w_8^3 & w_8^6 & w_8^1 & w_8^4 & w_8^7 & w_8^2 & w_8^5 \\ 1 & w_8^4 & 1 & w_8^4 & 1 & w_8^4 & 1 & w_8^4 \\ 1 & w_8^5 & w_8^2 & w_8^7 & w_8^4 & w_8^1 & w_8^6 & w_8^3 \\ 1 & w_8^6 & w_8^4 & w_8^2 & 1 & w_8^6 & w_8^4 & w_8^2 \\ 1 & w_8^7 & w_8^6 & w_8^5 & w_8^4 & w_8^3 & w_8^2 & w_8^1 \end{bmatrix}. \tag{4.21}$$

If we define the permutation matrix $\mathbf{M}_8$ with respect to the index vector $c = [\ 0\ 2\ 4\ 6\ 1\ 3\ 5\ 7\ ]$ then

$$\tilde{\mathbf{W}}_8 := \mathbf{W}_8 \cdot \mathbf{M}_8$$
$$= \left[\begin{array}{cccc|cccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & w_8^2 & w_8^4 & w_8^6 & w_8^1 & w_8^3 & w_8^5 & w_8^7 \\ 1 & w_8^4 & 1 & w_8^4 & w_8^2 & w_8^6 & w_8^2 & w_8^6 \\ 1 & w_8^6 & w_8^4 & w_8^2 & w_8^3 & w_8^1 & w_8^7 & w_8^5 \\ \hline 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & w_8^2 & w_8^4 & w_8^6 & -w_8^1 & -w_8^3 & -w_8^5 & -w_8^7 \\ 1 & w_8^4 & 1 & w_8^4 & -w_8^2 & -w_8^6 & -w_8^2 & -w_8^6 \\ 1 & w_8^6 & w_8^4 & w_8^2 & -w_8^3 & -w_8^1 & -w_8^7 & -w_8^5 \end{array}\right]. \tag{4.22}$$

Consider $\tilde{\mathbf{W}}_8$ as a 2×2 matrix with 4×4 blocks. Noting that $w_8^2 = w_4^1$ we see that

$$\tilde{\mathbf{W}}_8 = \left[\begin{array}{c|c} \mathbf{W}_4 & \mathbf{\Lambda}_4\mathbf{W}_4 \\ \hline \mathbf{W}_4 & -\mathbf{\Lambda}_4\mathbf{W}_4 \end{array}\right] \tag{4.23}$$

where

$$\mathbf{\Lambda}_4 := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & w_8^1 & 0 & 0 \\ 0 & 0 & w_8^2 & 0 \\ 0 & 0 & 0 & w_8^3 \end{bmatrix}.$$

It is easily verified that

$$\tilde{\mathbf{W}}_8 = \begin{bmatrix} \mathbf{W}_4 & \mathbf{\Lambda}_4\mathbf{W}_4 \\ \mathbf{W}_4 & -\mathbf{\Lambda}_4\mathbf{W}_4 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{I}_4 & \mathbf{\Lambda}_4 \\ \mathbf{I}_4 & -\mathbf{\Lambda}_4 \end{bmatrix} \begin{bmatrix} \mathbf{W}_4 & \mathbf{0}_4 \\ \mathbf{0}_4 & \mathbf{W}_4 \end{bmatrix} \tag{4.24}$$

$$= \begin{bmatrix} \mathbf{I}_4 & \mathbf{I}_4 \\ \mathbf{I}_4 & -\mathbf{I}_4 \end{bmatrix} \begin{bmatrix} \mathbf{W}_4 & \mathbf{0}_4 \\ \mathbf{0}_4 & \mathbf{\Lambda}_4\mathbf{W}_4 \end{bmatrix} \tag{4.25}$$

$$= (\mathbf{H}_2 \otimes \mathbf{I}_4) \begin{bmatrix} \mathbf{I}_4 & \mathbf{0}_4 \\ \mathbf{0}_4 & \mathbf{\Lambda}_4 \end{bmatrix} \begin{bmatrix} \mathbf{W}_4 & \mathbf{0}_4 \\ \mathbf{0}_4 & \mathbf{W}_4 \end{bmatrix} \tag{4.26}$$

$$= (\mathbf{H}_2 \otimes \mathbf{I}_4)(\mathbf{I}_4 \oplus \mathbf{\Lambda}_4)(\mathbf{I}_2 \otimes \mathbf{W}_4) \tag{4.27}$$

where $\mathbf{I}_4$ and $\mathbf{0}_4$ are the 4×4 unit matrix and the zero matrix, respectively. Replacing $\mathbf{W}_4$ with $\mathbf{Q}_4\mathbf{R}_4\mathbf{Q}_4$ and using (4.17), (4.18) we obtain

$$\tilde{\mathbf{W}}_8 = (\mathbf{H}_2 \otimes \mathbf{I}_4)(\mathbf{I}_4 \oplus \mathbf{\Lambda}_4)(\mathbf{I}_2 \otimes \mathbf{W}_4) \tag{4.28}$$

$$= (\mathbf{H}_2 \otimes \mathbf{I}_4)(\mathbf{I}_4 \oplus \mathbf{\Lambda}_4)(\mathbf{I}_2 \otimes (\mathbf{Q}_4\mathbf{R}_4\mathbf{Q}_4)). \tag{4.29}$$

Since the matrix $\mathbf{M}_8$ is a permutation matrix it yields $\mathbf{M}_8^T = \mathbf{M}_8^{-1}$ and hence

$$\begin{aligned} \mathbf{W}_8 &= \tilde{\mathbf{W}}_8\mathbf{M}_8^T \\ &= \underbrace{(\mathbf{H}_2 \otimes \mathbf{I}_4)}_{8\ add.}\ \underbrace{(\mathbf{I}_4 \oplus \mathbf{\Lambda}_4)}_{3\ mult.}\ \underbrace{(\mathbf{I}_2 \otimes (\mathbf{Q}_4\mathbf{R}_4\mathbf{Q}_4))}_{2 \times (8\ add. + 1\ mult.)}\ \mathbf{M}_8^T. \end{aligned} \tag{4.30}$$

The proposed algorithm (3.12) and Eq. (4.20) show that the number of arithmetic operations required in the FFT of order 8 is 24 additions ($N \log_2 N$, $N = 8$) and 5 multiplications. Now we define permutation matrix $\mathbf{M}_{16}$ with respect to the index vector $c = [0\ 2\ 4\ 6\ 8\ 10\ 12\ 14\ 1\ 3\ 5\ 7\ 9\ 11\ 13\ 15]$ and $\Lambda_8 := \mathrm{diag}(1,\ w_{16}^1,\ w_{16}^2,\ \ldots,\ w_{16}^7)$, then, we obtain in an analogous manner,

$$\tilde{\mathbf{W}}_{16} := (\mathbf{H}_2 \otimes \mathbf{I}_8)(\mathbf{I}_8 \oplus \mathbf{\Lambda}_8)(\mathbf{I}_2 \otimes \mathbf{W}_8) \tag{4.31}$$

where $\mathbf{I}_8$ and $\mathbf{0}_8$ are the 8×8 unit matrix and the zero matrix, respectively. Hence, we have

$$\begin{aligned} \mathbf{W}_{16} &= \tilde{\mathbf{W}}_{16}\,\mathbf{M}_{16}^T \\ &= \underbrace{(\mathbf{H}_2 \otimes \mathbf{I}_8)}_{16\ add.}\ \underbrace{(\mathbf{I}_8 \oplus \Lambda_8)}_{7\ mult.}\ \underbrace{(\mathbf{I}_2 \otimes \mathbf{W}_8)}_{48\,add. + 10\,mult.}\ \mathbf{M}_{16}^T. \end{aligned} \tag{4.32}$$

From the algorithm (3.12) and Eq. (4.20) we infer that the number of arithmetic operations required in the FFT of order 16 is 64 additions ($N \log_2 N$, $N = 8$) and 17 multiplications. In a similar way we have N-point FFT matrix $\mathbf{W}_N$ as follows:

$$\mathbf{W}_N = \tilde{\mathbf{W}}_N \mathbf{M}_N^T = \underbrace{(\mathbf{H}_2 \otimes \mathbf{I}_{N/2})}_{N\ add.} \underbrace{(\mathbf{I}_{N/2} \oplus \mathbf{\Lambda}_{N/2})}_{\frac{N}{2}-1\ mult.} \underbrace{(\mathbf{I}_2 \otimes \mathbf{W}_{N/2})}_{2\times Op.\ for\ \mathbf{W}_{N/2}} \mathbf{M}_N^T, \tag{4.33}$$

where $\mathbf{\Lambda}_{N/2} := \mathrm{diag}(1,\ w_N^1,\ w_N^2,\ \ldots,\ w_N^{N/2-1})$ and the corresponding index vector for the permutation matrix $\mathbf{M}_N$ is $c = [\,0\ 2\ \cdots\ N-2\ 1\ 3\ \cdots\ N-1\,]$. This is another way of representing FFT in a relatively simpler manner than the method in [3] since the second term of (4.33) is always a diagonal matrix and other terms are described only in terms of the Kronecker product and the direct sum.

5. Conclusions

The Reverse Jacket matrix as a generalizing Hadamard matrix was described. We have presented the 1-D fast Reverse Jacket transform algorithm and have shown that FRJT can be applied to represent the fast Fourier transform (FFT) in terms of matrix decomposition using 4×4 FRJT. Our method gives a clear representation of FFT in the sense that the second term of (4.33) is always a diagonal matrix and other terms are described only in terms of the Kronecker product and the direct sum. In addition, if we wish to put some weighting factors on the elements of the Hadamard matrix, only the elements of the diagonal matrix will be changed. However, understanding the complete relationship between the N-point FRJT and the N-point FFT needs further research. The Reverse Jacket matrix can be used together with the Hadamard matrix in many application areas such as filter banks, communications and signal/image processing, and orthogonal transforms and fast algorithms.

A. Appendix

In order to prove Theorem 2.3 and Corollary 2.5, we first give the following well-known lemma.

Lemma A.1 (*Matrix inversion lemma*). *Let*

$$\mathbf{M} := \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix},$$

where $\mathbf{M}_{11}$ *and* $\mathbf{M}_{22}$ *are* $n \times n$ *and* $m \times m$ *nonsingular submatrices, respectively, such that*

$$\left(\mathbf{M}_{11} - \mathbf{M}_{12}\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\right) \quad and \quad \left(\mathbf{M}_{22} - \mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{M}_{12}\right)$$

are also nonsingular. Then $\mathbf{M}$ *is nonsingular with*

$$\mathbf{M}^{-1} = \begin{bmatrix} \mathbf{M}_{11}^{-1} + \mathbf{M}_{11}^{-1}\mathbf{M}_{12}\big(\mathbf{M}_{22} - \mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{M}_{12}\big)\mathbf{M}_{21}\mathbf{M}_{11}^{-1} & -\mathbf{M}_{11}^{-1}\mathbf{M}_{12}\big(\mathbf{M}_{22} - \mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{M}_{12}\big)^{-1} \\ -\big(\mathbf{M}_{22} - \mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{M}_{12}\big)^{-1}\mathbf{M}_{21}\mathbf{M}_{11}^{-1} & \big(\mathbf{M}_{22} - \mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{M}_{12}\big)^{-1} \end{bmatrix}$$
$$= \begin{bmatrix} \big(\mathbf{M}_{11} - \mathbf{M}_{12}\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\big)^{-1} & -\big(\mathbf{M}_{11} - \mathbf{M}_{12}\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\big)^{-1}\mathbf{M}_{12}\mathbf{M}_{22}^{-1} \\ -\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\big(\mathbf{M}_{11} - \mathbf{M}_{12}\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\big)^{-1} & \mathbf{M}_{22}^{-1} + \mathbf{M}_{22}^{-1}\mathbf{M}_{21}\big(\mathbf{M}_{11} - \mathbf{M}_{12}\mathbf{M}_{22}^{-1}\mathbf{M}_{21}\big)^{-1}\mathbf{M}_{12}\mathbf{M}_{22}^{-1} \end{bmatrix}.$$

Proof of Theorem 2.3: According to Lemma A.1 and Definition 2.1, RJMs with higher order are always invertible if the 4×4 RJM is nonsingular. As in Lemma A.1 we assume that

$$\mathbf{N}_1 := \mathbf{R}_2 + \mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2^{-1}\,\mathbf{J}_2\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2 \quad \text{and}$$
$$\mathbf{Y}_1 := -\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 - \mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2 \tag{1.34}$$

are nonsingular, then it holds that for $k \geq 2$

$$\mathbf{N}_k := \mathbf{R}_{2^k} + \mathbf{Z}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{S}_{2^k}\,\mathbf{J}_{2^k}\,\mathbf{R}_{2^k}^{-1}\,\mathbf{J}_{2^k}\,\mathbf{S}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{Z}_{2^k}$$

and

$$\mathbf{Y}_k := -\mathbf{J}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{J}_{2^k} - \mathbf{S}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{Z}_{2^k}\,\mathbf{R}_{2^k}^{-1}\,\mathbf{Z}_{2^k}\,\mathbf{R}_{2^k}\,\mathbf{S}_{2^k}$$

are also nonsingular. It is easily verified that the matrices $\mathbf{N}_k$, $\mathbf{Y}_k$ for $k \geq 1$ are invertible if $a,\ b,\ c,\ d \neq 0$ and $\det \mathbf{R}_2 \neq 0$. The first assumption comes directly from Definition 2.1. Let us consider the case of RJM of order 4. The proofs for higher order RJMs can be carried out in an analogous manner. Letting

$$\mathbf{T}_1 := -\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 - \mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2$$

under the assumptions, and using the matrix inverse lemma (A.1), we write

$$\mathbf{R}_4{}^{-1} = \begin{bmatrix} \check{\mathbf{R}}_2^{-1} & -\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\,T_1^{-1} \\ -T_1^{-1}\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1} & T_1^{-1} \end{bmatrix}.$$

Hence, it is sufficient to prove that

(i) $-\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\,T_1^{-1} = \mathbf{Z}_2\,\check{\mathbf{R}}_2^{-1}\,\mathbf{S}_2$,
(ii) $-T_1^{-1}\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1} = \mathbf{S}_2\,\check{\mathbf{R}}_2^{-1}\,\mathbf{Z}_2$,
(iii) $T_1^{-1} = -\,\mathbf{J}_2\,\check{\mathbf{R}}_2^{-1}\,\mathbf{J}_2$.

Since $\mathbf{S}_2\,\mathbf{S}_2 = \mathbf{Z}_2\,\mathbf{Z}_2 = [\begin{smallmatrix}1 & 0\\ 0 & 1\end{smallmatrix}]$, $\mathbf{J}_2\,\mathbf{S}_2 = \mathbf{Z}_2$ and $\mathbf{S}_2\,\mathbf{J}_2 = -\mathbf{Z}_2$ it is true that

$$\begin{aligned} -T_1\,\mathbf{S}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2 &= -(-\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 - \mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2) \\ &\quad \times \mathbf{S}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2 \\ &= \mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2\,\mathbf{S}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2 \\ &\quad + \mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2 \\ &= \mathbf{S}_2\,\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{Z}_2 + \mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2 \\ &= \mathbf{S}_2\big(\mathbf{R}_2 + \mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{Z}_2\big)\mathbf{Z}_2 \\ &= \mathbf{S}_2\big(\mathbf{R}_2 + \mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2^{-1}\,\mathbf{J}_2\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\big)\mathbf{Z}_2 \\ &= \mathbf{S}_2\,\check{\mathbf{R}}_2\,\mathbf{Z}_2. \end{aligned}$$

Thus, it is true that

$$-\,T_1\,\mathbf{S}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2 = \mathbf{S}_2\,\check{\mathbf{R}}_2\,\mathbf{Z}_2. \qquad (1.35)$$

Taking the inverse on both sides of (1.35),

$$-\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\,T_1^{-1} = \mathbf{Z}_2\,\check{\mathbf{R}}_2^{-1}\,\mathbf{S}_2.$$

This proves (i). Now we will prove (ii).

$$\begin{aligned} -\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{S}_2\,T_1 &= -\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{S}_2 \\ &\quad \times \big(-\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 - \mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\big) \\ &= \mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 \\ &\quad + \mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{S}_2\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2 \\ &= \mathbf{Z}_2\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 \\ &\quad + \mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2 \\ &= \mathbf{Z}_2\big(\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 + \mathbf{R}_2\,\mathbf{S}_2\big) \\ &= \mathbf{Z}_2(\mathbf{R}_2 + \mathbf{Z}_2\,\mathbf{R}_2\,(-\,\mathbf{S}_2\,\mathbf{J}_2)\mathbf{R}_2^{-1}(-\mathbf{J}_2\,\mathbf{S}_2)\mathbf{R}_2\,\mathbf{J}_2\,\mathbf{S}_2)\mathbf{S}_2 \\ &= \mathbf{Z}_2(\mathbf{R}_2 + \mathbf{Z}_2\,\mathbf{R}_2\,(\mathbf{S}_2\,\mathbf{J}_2)\mathbf{R}_2^{-1}(\mathbf{J}_2\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2)\mathbf{S}_2 \\ &= \mathbf{Z}_2\,\check{\mathbf{R}}_2\,\mathbf{S}_2. \end{aligned}$$

Thus, it is true that

$$-\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{S}_2\,T_1 = \mathbf{Z}_2\,\check{\mathbf{R}}_2\,\mathbf{S}_2.$$

This implies that

$$-T_1^{-1}\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1} = \mathbf{S}_2\,\check{\mathbf{R}}_2^{-1}\,\mathbf{Z}_2.$$

This proves (ii). Finally, we obtain analogously

$$\begin{aligned} -\mathbf{J}_2\,\check{\mathbf{R}}_2\,\mathbf{J}_2 &= -\mathbf{J}_2\big(\mathbf{R}_2 + \mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2^{-1}\,\mathbf{J}_2\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\big)\mathbf{J}_2 \\ &= -\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 - \mathbf{J}_2\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2^{-1}\,\mathbf{J}_2\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{J}_2 \\ &= -\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 - \mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2\,\mathbf{R}_2^{-1}\,\mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2 \\ &= \mathbf{T}_1 \end{aligned}$$

This implies that

$$\mathbf{T}_1^{-1} = -\mathbf{J}_2\,\check{\mathbf{R}}_2^{-1}\,\mathbf{J}_2.$$

This proves (iii). Thus, (i), (ii), and (iii) complete the proof of the theorem. □

Proof of Corollary 2.5: Let $\mathbf{U}_{2^{k+1}} := [\begin{smallmatrix}\mathbf{I}_{2^k} & \mathbf{0}_{2^k}\\ \mathbf{0}_{2^k} & \mathbf{S}_{2^k}\end{smallmatrix}]$ where $\mathbf{I}_{2^k}$, $\mathbf{0}_{2^k}$ are the identity matrix and zero matrix with order 2^k for $k \geq 1$, respectively. We consider the case of order 4. The cases of higher orders can be proved analogously. It is easily checked that $\mathbf{Q}_4^T = \mathbf{Q}_4 = \mathbf{Q}_4^{-1}$ where T denotes the transpose of the matrix.

$$\begin{aligned} \mathbf{Q}_4 \cdot \mathbf{R}_4 \cdot \mathbf{Q}_4 &= \begin{bmatrix} \mathbf{I}_2 & \mathbf{O}_2 \\ \mathbf{O}_2 & \mathbf{S}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R}_2 & \mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2 \\ \mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2 & -\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 \end{bmatrix} \begin{bmatrix} \mathbf{I}_2 & \mathbf{O}_2 \\ \mathbf{O}_2 & \mathbf{S}_2 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{R}_2 & \mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2 \\ \mathbf{S}_2\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2 & -\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2 \end{bmatrix} \begin{bmatrix} \mathbf{I}_2 & \mathbf{O}_2 \\ \mathbf{O}_2 & \mathbf{S}_2 \end{bmatrix} \end{aligned}$$

$$= \begin{bmatrix} \mathbf{R}_2 & \mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{S}_2\,\mathbf{S}_2 \\ \mathbf{S}_2\,\mathbf{S}_2\,\mathbf{R}_2\,\mathbf{Z}_2 & -\mathbf{S}_2\,\mathbf{J}_2\,\mathbf{R}_2\,\mathbf{J}_2\,\mathbf{S}_2 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{R}_2 & \mathbf{Z}_2\,\mathbf{R}_2 \\ \mathbf{R}_2\,\mathbf{Z}_2 & \mathbf{Z}_2\,\mathbf{R}_2\,\mathbf{Z}_2 \end{bmatrix}$$

$$= \mathbf{X}_4.$$

This completes the proof of the Corollary. □

References

1. S.S. Agaian, *Hadamard Matrices and Their Applications*, Lecture Notes in Mathematics, Vol. 1168, 1985.
2. C.-P. Fan and J.-F. Yang, "Fast center-weighted Hadamard transform algorithms," *IEEE Trans. on Circuits and Systems-II*, Vol. 45, No. 3, pp. 429–432, 1998.
3. G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd edn., The Johns Hopkins University Press: Baltimore, 1996, pp. 1247–1249.
4. R.A. Horn and C.R. Johnson, *Topics in Matrix Analysis*, Cambridge Univ. Press: New York, 1991.
5. P. Lancaster and M. Tismenetsky, *The Theory of Matrices*, Computer Science and Applied Mathematics, Academic Press: San Diego, CA 1983.
6. M.H. Lee, "The Center-weighted Hadamard transform," *IEEE Trans. on Circuits and Systems*, Vol. 36, No. 9, pp. 1247–1249, 1989.
7. M.H. Lee, "A new reverse Jacket transform and its fast algorithm," *IEEE Trans. on Circuits and Systems-II*, Vol. 47, No. 1, pp. 39–47, 2000.
8. S.-R. Lee and M.H. Lee, "On the Reverse Jacket matrix for weighted Hadamard transform," *IEEE Trans. on Circuits and Systems-II*, Vol. 45, No. 3, pp. 436–441, 1998.
9. S.-R. Lee and W.H. Kwon, "The Reverse Jacket matrix as a generalized Hadamard matrix," Haedong Digital Library, Seoul National University, Technical Report SNU-EE-TR-1999-4, 1999.
10. F.J. Macwilliams and N.J. Sloane, *The Theory of Error-Correcting Codes*, North-Holland: Amesterdam, 1992.
11. A.V. Oppenheim and R.W. Schafer, *Discrete-Time Signal Processing*, Prentice-Hall: New York, 1989.
12. M.G. Parker and M.H. LEE, "Optimal bipolar sequences for the complex Reverse-Jacket transform," in *Int. Symp. on Information Theory and Its Applications*, Honolulu, Hawaii, USA, Nov. 5–8, 2000.
13. S. Roman, *Advanced Linear Algebra*, Graduate Texts in Mathematics, Vol. 135, Springer Verlag: Berlin, 1992.
14. R.K.R. Yarlagadda, *Hadamard Matrix Analysis and Synthesis*, Kluwer Academic Publishers: Dordrecht, 1997.

Seung-Rae Lee received the B.S. degree in 1991, M.S. degree in 1993, and doctoral degree in 1997, all in mathematics from the University of Technology Aachen, Germany. From 1995 to 1996 he was an assistant in mathematics at the Gerhard-Mercator University of Duisburg, Germany. From 1998 to 2000, he was a postdoctoral researcher with the Automation and Systems Research Institute at Seoul National University. In 2000, he was a research professor with the School of Electrical and Computer Engineering at Sungkyunkwan University. He is currently a senior researcher with the Institute of New Media and Communication at Seoul National University, Korea. His current research interests are differential games, optimal control, signal/image processing and acoustic engineering. He is a member of the Korean Mathematical Society, Korean Institute of Communication Sciences, and Acoustical Society of Korea.

June-Ho Yi received the B.S. degree from the Seoul National University, Seoul, Korea, in 1985, the M.S. degree from the Pennsylvania State University, University Park, PA, in 1987, and the Ph.D. degree from Purdue University in 1994, all in Electrical and Computer Engineering.

In 1989, he was a research scientist at the Samsung Advanced Institute of Technology. From 1994 to 1995, he was a research scientist at the University of California, Riverside. From 1995 to 1996, he was a senior research scientist at the Korea Institute of Science and Technology, Seoul, Korea. In 1997, he joined Sungkyunkwan University, Korea, and he is currently an associate professor of School of Electrical and Computer Engineering. His research interests are broadly in the areas of computer vision and vision-based computer interaction.