

Matrix Factorizations for Reversible Integer Mapping

Pengwei Hao, *Member, IEEE*, and Qingyun Shi

Abstract—Reversible integer mapping is essential for lossless source coding by transformation. A general matrix factorization theory for reversible integer mapping of invertible linear transforms is developed in this paper. Concepts of the integer factor and the elementary reversible matrix (ERM) for integer mapping are introduced, and two forms of ERM—triangular ERM (TERM) and single-row ERM (SERM)—are studied. We prove that there exist some approaches to factorize a matrix into TERMs or SERMs if the transform is invertible and in a finite-dimensional space. The advantages of the integer implementations of an invertible linear transform are i) mapping integers to integers, ii) perfect reconstruction, and iii) in-place calculation. We find that besides a suitable permutation matrix, the TERM factorization of an N -by- N nonsingular matrix has at most three TERMs, and its SERM factorization has at most $N + 1$ SERMs. The elementary structure of ERM transforms is the ladder structure. An executable factorization algorithm is also presented. Then, the computational complexity is compared, and some optimization approaches are proposed. The error bounds of the integer implementations are estimated as well. Finally, three ERM factorization examples of DFT, DCT, and DWT are given.

Index Terms—Lifting scheme, linear transforms, lossless compression, matrix factorization, reversible integer mapping.

I. INTRODUCTION

TO COMPRESS digital signals losslessly by means of transformation, the transform must map integers to integers and be perfectly invertible. A unified lossy/lossless compression system, which enables region-of-interest (ROI)-based image coding and progressive quality from the coarsest levels to the finer levels or even the lossless level, also demands a lossless integer transform. Perfect inversion or reversible integer mapping is, therefore, significant for source coding.

For a long time, people devised one-to-one integer mapping approaches for some simple transforms, such as the S transform [1], the TS transform [2], the $S + P$ transform [3], and the color space transforms [4]. In 1992, Bruickers and van den Eenden proposed a ladder structure of filtering networks in filterbank design for perfect inversion and perfect reconstruction [5]. Their work opened a new path for integer transform research. However, more research needs to be done along this line. Initially, integer transforms were studied systematically using similar networks in the research on discrete wavelet transforms, where the previously mentioned networks are utilized to construct wavelets

and to increase the vanishing moment. The method is called the lifting scheme by Sweldens [6]. Dewitte and Cornelis applied the lifting scheme to implement wavelet transforms by integer mapping [7]. Later, Daubechies and Sweldens *et al.* in [8] and [9] and Shi in [10] extended the lifting scheme to general wavelet transforms. They presented some schemes that can convert discrete wavelet transform into integer transform steps. Their work has been widely accepted and adopted. Recently, Hao and Shi in [11] proved that there exists an implementation of integer mapping (including a scaling modification if necessary) when a linear transform is invertible and in finite-dimensional space, but the factorization is not optimal.

In this paper, an optimal solution of reversible integer implementation for general linear transforms is presented. It is a generalized lifting scheme and can be applied to any invertible linear transform besides discrete wavelet transforms. Although wavelet transforms gained general acceptance in the signal coding community, there is still some demand for reversible integer implementation for some other transforms such as color space transforms. In fact, based on our matrix factorization theory, we submitted two proposals [12], [13] on the reversible multiple component transform for the emerging image coding standard JPEG 2000. More importantly, we presented a general theory of reversible integer mapping for linear transforms, which may evoke some other methods and applications.

The following will suggest what we can do to solve the problem. An affine transform in finite-dimensional space can be expressed in a matrix form $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b}$, which becomes a linear transform when $\mathbf{b} = \mathbf{0}$. If it directly maps integers to integers by rounding arithmetic (denoted by brackets $[\]$, $\hat{\mathbf{y}} = [\mathbf{A}\mathbf{x} + \mathbf{b}]$), rounding errors may be inevitable: $\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - [\mathbf{A}\mathbf{x} + \mathbf{b}] = \mathbf{A}\mathbf{x} + \mathbf{b} - [\mathbf{A}\mathbf{x} + \mathbf{b}]$. For the uncertain rounding error vector, \mathbf{e} , $\|\mathbf{e}\|_\infty \leq u$, where u is the unit roundoff, which is defined as the largest error that can occur in a rounding or chopping operation. Then, we have $[\mathbf{e}] = \mathbf{0}$. To simplify our depictions, the arithmetic rounding operation is considered to be integer-independent, or $[c + n] = [c] + n$, where n is an integer. Thus, if the transform matrix \mathbf{A} is nonsingular, then the perfectly reversible condition will be $\mathbf{x} = [\mathbf{A}^{-1}(\hat{\mathbf{y}} - \mathbf{b})] = [\mathbf{A}^{-1}([\mathbf{A}\mathbf{x} + \mathbf{b}] - \mathbf{b})] = [\mathbf{A}^{-1}(\mathbf{A}\mathbf{x} + \mathbf{b} - \mathbf{e} - \mathbf{b})] = [\mathbf{x} - \mathbf{A}^{-1}\mathbf{e}] = \mathbf{x} + [-\mathbf{A}^{-1}\mathbf{e}]$, or $[-\mathbf{A}^{-1}\mathbf{e}] = \mathbf{0}$. Thus, its necessary and sufficient condition is that $\|-\mathbf{A}^{-1}\mathbf{e}\|_\infty \leq u$. Since $\|-\mathbf{A}^{-1}\mathbf{e}\|_\infty = \|\mathbf{A}^{-1}\mathbf{e}\|_\infty \leq \|\mathbf{A}^{-1}\|_\infty \cdot \|\mathbf{e}\|_\infty \leq \|\mathbf{A}^{-1}\|_\infty \cdot u$, a sufficient condition for $\|-\mathbf{A}^{-1}\mathbf{e}\|_\infty \leq u$ is $\|\mathbf{A}^{-1}\|_\infty \leq 1$. Perfect inversion of a transform demands that both the forward transform and its reverse are integer reversible. Analogously, if the transformed values are changed due to quantization or some other operation, we still anticipate that the values can be reversely transformed into integers. Then, the sufficient integer reversible condition for the inverse transform is $\|\mathbf{A}\|_\infty \leq 1$. However, because $\|\mathbf{A}\|_\infty \|\mathbf{A}^{-1}\|_\infty \geq \|\mathbf{A}\mathbf{A}^{-1}\|_\infty = 1$, both

Manuscript received April 21, 2000; revised June 8, 2001. This work was supported by NKBRSF China under Grant G1998030606 and by funding for recipients of National Excellent Doctoral Dissertation, China. The associate editor coordinating the review of this paper and approving it for publication was Prof. Paulo S. R. Diniz.

The authors are with the Center for Information Science, National Laboratory on Machine Perception, Peking University, Beijing, China (e-mail: phao@cis.pku.edu.cn; sqy@pku.edu.cn).

Publisher Item Identifier S 1053-587X(01)07774-1.

conditions cannot be satisfied at the same time, except for some elementary matrices of $\|\mathbf{A}\|_\infty = \|\mathbf{A}^{-1}\|_\infty = 1$. A transform matrix does not always satisfy this condition, but those called the elementary reversible matrices (ERMs) can. Therefore, we concentrate on

- i) whether a transform matrix can be factorized into a series of elementary integer reversible transform matrices;
- ii) how to find this kind of factorization;
- iii) what its optimal form is.

Because factorizing a transform matrix into a series of ERMs is the basic approach for implementing the transform by integer mapping, this paper addresses the matrix factorization theory of general linear transforms. In Section II, some definitions and preliminaries are given, and the elementary structures of integer linear transforms and the forms of ERMs are studied. It is also pointed out that the ladder structure is the proper structure for reversible integer mapping. In Section III, two types of elementary reversible matrix factorization—triangular ERM (TERM) and single-row ERM (SERM)—are proposed and theoretically proved. Detailed factorization algorithms are given in Section IV. Section V shows the implementation structures of the linear transforms. The computational complexity of the transform implementations is compared, and optimization suggestions for a specific matrix are given in Section VI. By contrast with the symbolic or theoretical infinite-precision calculation, an integer transform produces errors as a result of arithmetic rounding operations; therefore, the upper bounds of such errors are also discussed in Section VII. Then, in Section VIII, three typical examples of the 4×4 DFT matrix, the 8×8 DCT matrix, and Harr wavelet polyphase matrix factorizations are given to show the availability and practicability of reversible integer mapping. Conclusions are given at the end of the paper.

II. DEFINITIONS AND PRELIMINARIES

In the following discussion, a linear transform is defined in the complex number field. An integer referred to below in the complex number field is considered a complex number whose real and imaginary components are integers. An arithmetic rounding operation for a complex number implies an integer conversion for both its real and imaginary parts.

For descriptive convenience, an **integer factor** is defined as a multiplier of integers that does not change their magnitude, which is denoted by j . Obviously, its reciprocal $1/j$ is also an integer factor. For real numbers, an integer factor can be 1 or -1 ; for complex integers, integer factors may be 1, -1 , i , or $-i$, where i is the imaginary unit.

Many matrix computation algorithms that map number x to number y , such as those for a linear transform, can be calculated in a computing machinery like the “saxpy” [14] mechanism, $y = ax + b$. If variable x can be assigned to an arbitrary integer, the condition of integer output y is that both number a and number b are integers. Since there might be some processing procedures in the transformation domain, the range of y is considered to be a set of integers without restrictions. In the case of an arbitrary integer output, the condition for the inverse transform that maps integers to integers is that both $1/a$ and b are integers. Therefore,

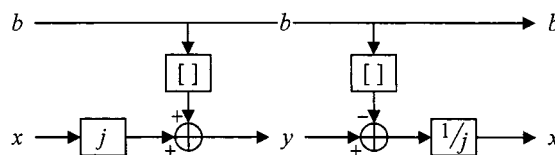


Fig. 1. Forward and reverse transform of a number implemented by integer mapping.

the perfectly integer-invertible condition of the transform is that a is an integer factor and b is an integer.

If b is not an integer, the computations may still be done with floating-point numbers, but an integer representation $[b]$ can be a substitution. Then, the transform result is guaranteed to be an integer, and its invertibility is preserved. The integer conversion methods can be rounding (by C function, round), chopping or rounding-down (floor), rounding-up (ceil), or some other methods.

It follows that a form of an invertible transform of scalar numbers implemented by integer mapping is $y = jx + [b]$, and its inversion $x = (1/j) \cdot (y - [b])$. Its flow chart is illustrated in Fig. 1.

The chart’s structure allows

- i) mapping integers to integers;
- ii) perfect reconstruction;
- iii) calculating in-place and without allocating auxiliary memory.

It is also an elementary structure of the integer implementation. For reversible integer transforms, it is necessary for number b to satisfy both of the following conditions.

- i) For forward transform calculation, b does *not* depend on x , either directly or indirectly.
- ii) For reverse transform calculation, b does *not* depend on y , either directly or indirectly.

Either in the real or in the complex number field, the above structure is a kind of ladder structure. In the case of $j = 1$, the structure is the same as the ladder structure proposed by Bruekers and van den Enden [5] and the lifting scheme structure by Sweldens [6]. Therefore, we can say that the ladder structure is the appropriate elementary structure for reversible integer mapping.

If the general N -dimensional invertible linear transform $\mathbf{y} = \mathbf{A}\mathbf{x}$ allows perfectly invertible integer mapping, its corresponding elementary computational structures must meet the two necessary conditions for number b mentioned above. If the computational ordering is properly arranged, we can find some elementary reversible structures for perfectly invertible integer implementation. A corresponding matrix is defined as an **elementary reversible matrix (ERM)**. An upper or lower triangular transform matrix whose diagonal elements are integer factors is a kind of ERM, which is called a **triangular ERM (TERM)**. If all the diagonal elements of a TERM are equal to 1, the TERM will be a unit triangular matrix.

A TERM has the following two important properties.

- i) The product of two upper TERMS is also an upper TERM, and the product of two lower TERMS also makes a lower TERM.
- ii) The determinant of a TERM is an integer factor.

If $\mathbf{A} = \{a_{mn}\}$ is an upper TERM, the computational ordering of linear transform $\mathbf{y} = \mathbf{A}\mathbf{x}$ can be arranged to be top-down:

$$\begin{cases} y_m = j_m x_m + \left[\sum_{n=m+1}^N a_{mn} x_n \right] = j_m x_m + [b_m] \\ y_N = j_N x_N \end{cases} \quad (1)$$

where $m = 1, 2, 3, \dots, N-1$.

Its inverse ordering is reversed:

$$\begin{cases} x_N = y_N / j_N \\ x_m = (1/j_m) \cdot \left(y_m - \left[\sum_{n=m+1}^N a_{mn} x_n \right] \right) \\ = (1/j_m) \cdot (y_m - [b_m]) \end{cases} \quad (2)$$

where $m = N-1, N-2, \dots, 1$.

If \mathbf{A} is a lower TERM, the computational ordering of linear transform $\mathbf{y} = \mathbf{A}\mathbf{x}$ can be arranged to be bottom-up and its inversion to be top-down:

$$\begin{cases} y_m = j_m x_m + \left[\sum_{n=1}^{m-1} a_{mn} x_n \right] = j_m x_m + [b_m] \\ y_1 = j_1 x_1 \end{cases} \quad (3)$$

where $m = N, N-1, N-2, \dots, 2$

$$\begin{cases} x_1 = y_1 / j_1 \\ x_m = (1/j_m) \cdot \left(y_m - \left[\sum_{n=1}^{m-1} a_{mn} x_n \right] \right) \\ = (1/j_m) \cdot (y_m - [b_m]) \end{cases} \quad (4)$$

where $m = 2, 3, \dots, N$.

Obviously, if a matrix can be converted into a TERM using row and column permutation only, it is also an ERM. Then, we can find another feasible ERM form known as *single-row ERM (SERM)* with integer factors on the diagonal and only one row of off-diagonal elements that are not all zeros. For $m = 1, 2, 3, \dots, N$, a unit SERM can be formulated as $\mathbf{S}_m = \mathbf{I} + \mathbf{e}_m \mathbf{s}_m^T$, where \mathbf{e}_m is the m th standard basis vector formed as the m th column of the identity matrix, and \mathbf{s}_m is a vector whose m th element is 0. The unit SERMs share an important property: $\mathbf{S}_m^{-1} = \mathbf{I} - \mathbf{e}_m \mathbf{s}_m^T$. For convenience, we will use $\mathbf{S}_0 = \mathbf{I} + \mathbf{e}_N \mathbf{s}_0^T$ in the following discussion.

III. MATRIX FACTORIZATIONS

For an arbitrary N -by- N invertible matrix \mathbf{A} , if $|\det \mathbf{A}| = \alpha \neq 1$, the matrix can be modified to satisfy $|\det \mathbf{A}| = 1$. Regardless of the physical equivalents, the modification method can be scaling a division of $m (\leq N)$ constants $\alpha_1, \alpha_2, \dots, \alpha_m$, respectively, for the m rows of \mathbf{A} . The constants must satisfy

$$\prod_{k=1}^m |\alpha_k| = |\det \mathbf{A}| = \alpha. \quad (5)$$

An advantage of the above modification is that the dynamic ranges of the transformed components can be controlled, which

is very helpful for some applications apart from the physical equivalents of a transform, such as lossless compression.

Taking the physical equivalents into account, the relative magnitude of the transform eigenvalues cannot be changed. Therefore, a proportional scaling modification is the alternative, i.e., to let the scaling factors be $\alpha_1 = \alpha_2 = \dots = \alpha_N = \sqrt[N]{\alpha}$.

For N -by- N matrix \mathbf{A} and $|\det \mathbf{A}| = 1$, the following theorems can be derived.

Theorem 1: Matrix \mathbf{A} has a TERM factorization of $\mathbf{A} = \mathbf{P}\mathbf{V}_1\mathbf{V}_2 \cdots \mathbf{V}_M\mathbf{D}_R$ if and only if $|\det \mathbf{A}| = 1$, where M is finite, \mathbf{V}_k ($k = 1, 2, \dots, M$) are unit TERMS, \mathbf{P} is a permutation matrix, and \mathbf{D}_R is a rotator for only one complex number.

The proof was given in our paper [10] (See the Appendix). As pointed out in the paper, M is not larger than 7. In this paper, we will show that $M \leq 3$ (see Theorem 3 and Corollary 3).

Corollary 1: Matrix \mathbf{A} has a TERM factorization of $\mathbf{A} = \mathbf{P}\mathbf{V}_1\mathbf{V}_2 \cdots \mathbf{V}_M$ if and only if $\det \mathbf{A}$ is an integer factor.

Corollary 2: Matrix \mathbf{A} has a unit TERM factorization of $\mathbf{A} = \mathbf{P}\mathbf{V}_1\mathbf{V}_2 \cdots \mathbf{V}_M$ if and only if $\det \mathbf{A} = \det \mathbf{P} = \pm 1$.

If $\det \mathbf{A} = e^{i\theta}$ is not an integer factor, a complex rotation can be implemented with the real and imaginary components of a complex number. A two-dimensional (2-D) rotation matrix can be factorized into three-unit TERMS, such as [9]

$$\begin{aligned} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} &= \begin{bmatrix} 1 & 0 \\ (1 - \cos \theta) / \sin \theta & 1 \end{bmatrix} \begin{bmatrix} 1 & -\sin \theta \\ 0 & 1 \end{bmatrix} \\ &\cdot \begin{bmatrix} 1 & 0 \\ (1 - \cos \theta) / \sin \theta & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & (\cos \theta - 1) / \sin \theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \sin \theta & 1 \end{bmatrix} \\ &\cdot \begin{bmatrix} 1 & (\cos \theta - 1) / \sin \theta \\ 0 & 1 \end{bmatrix}. \end{aligned} \quad (6)$$

Therefore, the case of $\det \mathbf{A} = j$ is considered hereinafter if it is not clearly specified in context.

Lemma 1: There exists a permutation matrix \mathbf{P} such that the leading principal submatrices of $\mathbf{P}^T \mathbf{A}$ are all nonsingular if and only if \mathbf{A} is invertible (nonsingular).

Its necessity is obvious. Its sufficiency follows by applying Gaussian elimination with partial pivoting.

Lemma 2: Matrix \mathbf{A} has a factorization of $\mathbf{A} = \mathbf{P}\mathbf{L}\mathbf{D}\mathbf{U}$ if the leading principal submatrices of $\mathbf{P}^T \mathbf{A}$ are all nonsingular, where \mathbf{L} and \mathbf{U} are unit lower and unit upper triangular matrices, respectively, and \mathbf{D} is diagonal.

Its proof can be found in the book by Golub and van Loan [14, p. 97, Th. 3.2.1].

Suppose $\mathbf{D} = \text{diag}(d_1, d_2, \dots, d_N)$. For $k = 1, 2, 3, \dots, N$, the minor of the k th leading principal submatrix $\det(\mathbf{P}^T \mathbf{A})_k = d_1 d_2 \cdots d_k$. Let all the diagonal entries of \mathbf{D} be 1s or let \mathbf{D} be the identity matrix. Then, using Lemmas 1 and 2, we can obtain the following triangular factorization theorem.

Theorem 2: Matrix \mathbf{A} has a unit triangular factorization of $\mathbf{A} = \mathbf{L}\mathbf{U}$ if and only if the minors of the leading principal submatrices of \mathbf{A} are all 1s.

In general cases, we have another TERM factorization theorem, as follows.

Theorem 3: Given a nonsingular diagonal matrix $\mathbf{D}_R = \text{diag}(1, 1, \dots, 1, e^{i\theta})$, matrix \mathbf{A} has a factorization of $\mathbf{A} = \mathbf{PLD}_R\mathbf{US}_0$ if and only if $\det \mathbf{P}^T \mathbf{A} = \det \mathbf{D}_R \neq 0$.

Proof: Its necessity is obvious. $\det \mathbf{P}^T \mathbf{A} = \det(\mathbf{LD}_R\mathbf{S}_0) = \det \mathbf{D}_R \neq 0$.

A proof of its sufficiency is given as follows.

Suppose $\mathbf{S}_0 = \mathbf{I} + e_N \mathbf{s}_0^T = \mathbf{I} + e_N \cdot [s_1, s_2, \dots, s_{N-1}, 0]$, and $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_N]$. Then, we have $\mathbf{AS}_0^{-1} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_N] \cdot (\mathbf{I} - e_N \mathbf{s}_0^T) = [\mathbf{a}_1 - s_1 \mathbf{a}_N, \mathbf{a}_2 - s_2 \mathbf{a}_N, \dots, \mathbf{a}_{N-1} - s_{N-1} \mathbf{a}_N, \mathbf{a}_N]$.

For a given diagonal matrix \mathbf{D}_R , the resulting problem is that if there exists a SERM (also a TERM) \mathbf{S}_0 such that the minor of the k th leading principal submatrix of $\mathbf{P}^T \mathbf{AS}_0^{-1}$ is $\det(\mathbf{P}^T \mathbf{AS}_0^{-1})_k = 1$, where $k = 1, 2, 3, \dots, N-1$

$$\begin{aligned} \det(\mathbf{AS}_0^{-1})_k &= \det[\mathbf{a}_1 - s_1 \mathbf{a}_N, \mathbf{a}_2 - s_2 \mathbf{a}_N, \dots, \mathbf{a}_k - s_k \mathbf{a}_N]_k \\ &= \det[\mathbf{a}_1 - s_1 \mathbf{a}_N, \mathbf{a}_2 - s_2 \mathbf{a}_N, \dots, \mathbf{a}_k]_k \\ &\quad - s_k \det[\mathbf{a}_1 - s_1 \mathbf{a}_N, \mathbf{a}_2 - s_2 \mathbf{a}_N, \dots \\ &\quad \quad \quad \mathbf{a}_{k-1} - s_{k-1} \mathbf{a}_N, \mathbf{a}_N]_k \\ &= \det[\mathbf{a}_1 - s_1 \mathbf{a}_N, \mathbf{a}_2 - s_2 \mathbf{a}_N, \dots, \mathbf{a}_k]_k \\ &\quad - s_k \det[\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{k-1}, \mathbf{a}_N]_k. \end{aligned}$$

For the nonsingular matrix \mathbf{A} , there must exist a permutation matrix \mathbf{P} such that the leading principal submatrices of $\mathbf{P}^T [\mathbf{a}_N, \mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{N-1}]$ are all nonsingular, which can guarantee that $\det(\mathbf{P}^T \cdot [\mathbf{a}_N, \mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{k-1}])_k \neq 0$ or $\det(\mathbf{P}^T \cdot [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{k-1}, \mathbf{a}_N])_k \neq 0$ for $k = 1, 2, 3, \dots, N-1$.

Then, let

$$\begin{aligned} \det(\mathbf{P}^T \mathbf{AS}_0^{-1})_k &= \det(\mathbf{P}^T \cdot [\mathbf{a}_1 - s_1 \mathbf{a}_N, \mathbf{a}_2 - s_2 \mathbf{a}_N, \dots, \mathbf{a}_k])_k \\ &\quad - s_k \det(\mathbf{P}^T \cdot [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{k-1}, \mathbf{a}_N])_k = 1 \end{aligned}$$

for the unknown variable s_k , and the equation definitely has a solution of

$$s_k = \frac{\det(\mathbf{P}^T \cdot [\mathbf{a}_1 - s_1 \mathbf{a}_N, \mathbf{a}_2 - s_2 \mathbf{a}_N, \dots, \mathbf{a}_k])_k - 1}{\det(\mathbf{P}^T \cdot [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{k-1}, \mathbf{a}_N])_k} \quad (7)$$

where $k = 1, 2, 3, \dots, N-1$.

Therefore, under the condition of $\det \mathbf{P}^T \mathbf{A} = \det \mathbf{D}_R = e^{i\theta}$, all s_k exist. Then, we can obtain a SERM \mathbf{S}_0 such that $\mathbf{P}^T \mathbf{AS}_0^{-1} = \mathbf{LD}_R\mathbf{U}$ or $\mathbf{A} = \mathbf{PLD}_R\mathbf{US}_0$.

The proof is completed. \blacksquare

The proven factorization for $\det \mathbf{A} = e^{i\theta}$ shows that except for a rotation and a permutation transform for an invertible linear transform, there are at most three elementary reversible transforms, which is just what we need.

Corollary 3: Matrix \mathbf{A} has a TERM factorization of $\mathbf{A} = \mathbf{PLUS}_0$ if and only if $\det \mathbf{A} = j$ is an integer factor, where \mathbf{L} and \mathbf{U} are lower and upper TERMS, respectively.

Corollary 4: Matrix \mathbf{A} has a unit TERM factorization of $\mathbf{A} = \mathbf{PLUS}_0$ if and only if $\det \mathbf{A} = \det \mathbf{P} = \pm 1$, where \mathbf{L} and \mathbf{U} are unit lower and unit upper TERMS, respectively.

Corollaries 3 and 4 can, respectively, be proved with Theorem 3 by assigning $e^{i\theta}$ to be an integer factor and 1.

Theorem 4: Matrix \mathbf{A} has a unit SERM factorization of $\mathbf{A} = \mathbf{S}_N \mathbf{S}_{N-1} \cdots \mathbf{S}_1$ if and only if the minors of the leading principal submatrices of \mathbf{A} are all 1s, where \mathbf{S}_m ($m = 1, 2, \dots, N$) are unit SERMs.

Proof: Its necessity is obvious. A proof for its sufficiency is given below.

As all the minors of the leading principal submatrices are 1s, matrix \mathbf{A} can be factorized into two unit triangular matrices (Theorem 2) $\mathbf{A} = \mathbf{LU}$. Suppose that matrix \mathbf{A} is a partitioned matrix. Then, we have

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{A}_{21} \mathbf{A}_{11}^{-1} & \mathbf{A}_{22} - \mathbf{A}_{21} \mathbf{A}_{11}^{-1} \mathbf{A}_{12} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (8) \end{aligned}$$

where $\det \mathbf{A}_{11} = 1$, and the two factorized block matrices do not change the minors of each other's leading principal submatrices, which are also all 1s.

Hence, the following factorization can be carried through:

$$\begin{aligned} \mathbf{A} = \mathbf{LU} &= (\mathbf{LUS}_1^{-1}) \mathbf{S}_1 = (\mathbf{LUS}_1^{-1} \mathbf{S}_2^{-1}) \mathbf{S}_2 \mathbf{S}_1 = \cdots \\ &= (\mathbf{LUS}_1^{-1} \mathbf{S}_2^{-1} \cdots \mathbf{S}_{N-1}^{-1}) \mathbf{S}_{N-1} \cdots \mathbf{S}_2 \mathbf{S}_1 = \mathbf{S}_N \cdots \mathbf{S}_2 \mathbf{S}_1 \quad (9) \end{aligned}$$

where the m th row of \mathbf{S}_m is equal to the m th row of $\mathbf{LUS}_1^{-1} \mathbf{S}_2^{-1} \cdots \mathbf{S}_{m-1}^{-1}$, $m = 1, 2, 3, \dots, N$.

The proof is completed. \blacksquare

A unit TERM factorization can be refactorized into N unit SERMs. Therefore, Theorems 2 and 4 are equivalent.

Corollary 5: Matrix \mathbf{A} has a SERM factorization of $\mathbf{A} = \mathbf{PS}_N \mathbf{S}_{N-1} \cdots \mathbf{S}_1 \mathbf{S}_0$ if and only if $\det \mathbf{A} = j$ is an integer factor, where \mathbf{S}_m ($m = 0, 1, 2, \dots, N$) are SERMs.

Corollary 6: Matrix \mathbf{A} has a unit SERM factorization of $\mathbf{A} = \mathbf{PS}_N \mathbf{S}_{N-1} \cdots \mathbf{S}_1 \mathbf{S}_0$ if and only if $\det \mathbf{A} = \det \mathbf{P} = \pm 1$, where \mathbf{S}_m ($m = 0, 1, 2, \dots, N$) are unit SERMs.

Corollaries 5 and 6 can be easily verified by applying the corresponding theorems above.

Thus, we have proved that an arbitrary finite-dimensional invertible linear transform can be implemented by reversible integer mapping, or its modified transform matrix can be factorized into at most three TERMS or a series of SERMs. Every transform that consists of TERMS or SERMs is immediately integer reversible.

IV. MATRIX FACTORIZATION ALGORITHMS

It has been proved that a nonsingular matrix can be factorized into a product of at most three TERMS or a series of SERMs, but the proofs do not give executable algorithms. An algorithm of matrix factorization for reversible integer mapping is given herein below.

Suppose

$$\mathbf{A} = \begin{bmatrix} a_{1,1}^{(0)} & a_{1,2}^{(0)} & \cdots & a_{1,N}^{(0)} \\ a_{2,1}^{(0)} & a_{2,2}^{(0)} & \cdots & a_{2,N}^{(0)} \\ \cdots & \cdots & \cdots & \cdots \\ a_{N,1}^{(0)} & a_{N,2}^{(0)} & \cdots & a_{N,N}^{(0)} \end{bmatrix}. \quad (10)$$

Then, there must exist a permutation matrix \mathbf{P}_1 for row interchanges such that

$$\mathbf{P}_1 \mathbf{A} = \begin{bmatrix} p_{1,1}^{(1)} & p_{1,2}^{(1)} & \cdots & p_{1,N}^{(1)} \\ p_{2,1}^{(1)} & p_{2,2}^{(1)} & \cdots & p_{2,N}^{(1)} \\ \cdots & \cdots & \cdots & \cdots \\ p_{N,1}^{(1)} & p_{N,2}^{(1)} & \cdots & p_{N,N}^{(1)} \end{bmatrix} \quad (11)$$

and $p_{1,N}^{(1)} \neq 0$, and hence, there must be a number s_1 such that $p_{1,1}^{(1)} - s_1 \cdot p_{1,N}^{(1)} = 1$. Then, we can get $s_1 = (p_{1,1}^{(1)} - 1)/p_{1,N}^{(1)}$ and obtain a product of

$$\begin{aligned} \mathbf{P}_1 \mathbf{A} \mathbf{S}_{0,1} &= \mathbf{P}_1 \mathbf{A} \begin{bmatrix} 1 & & & \\ & \mathbf{I} & & \\ -s_1 & \mathbf{0} & & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & p_{1,2}^{(1)} & \cdots & p_{1,N}^{(1)} \\ p_{2,1}^{(1)} - s_1 p_{2,N}^{(1)} & p_{2,2}^{(1)} & \cdots & p_{2,N}^{(1)} \\ \cdots & \cdots & \cdots & \cdots \\ p_{N,1}^{(1)} - s_1 p_{N,N}^{(1)} & p_{N,2}^{(1)} & \cdots & p_{N,N}^{(1)} \end{bmatrix}. \end{aligned} \quad (12)$$

Then, the forward elimination of the first column can be achieved by multiplying an elementary Gauss matrix \mathbf{L}_1

$$\begin{aligned} \mathbf{L}_1 \mathbf{P}_1 \mathbf{A} \mathbf{S}_{0,1} &= \begin{bmatrix} 1 & & & \\ s_1 p_{2,N}^{(1)} - p_{2,1}^{(1)} & 1 & & \\ \cdots & & \mathbf{I} & \\ s_1 p_{N,N}^{(1)} - p_{N,1}^{(1)} & & & 1 \end{bmatrix} \mathbf{P}_1 \mathbf{A} \mathbf{S}_{0,1} \\ &= \begin{bmatrix} 1 & a_{1,2}^{(2)} & \cdots & a_{1,N}^{(2)} \\ 0 & a_{2,2}^{(2)} & \cdots & a_{2,N}^{(2)} \\ \cdots & \cdots & \cdots & \cdots \\ 0 & a_{N,2}^{(2)} & \cdots & a_{N,N}^{(2)} \end{bmatrix}. \end{aligned} \quad (13)$$

Continuing in this way, for $k = 2, 3, \dots, N-1$, \mathbf{P}_k defines the row interchanges among the k th through the N th rows to guarantee that the k th element in the N th column are not zero $p_{k,N}^{(k)} \neq 0$. (If there were no such element, \mathbf{A} would have been singular.) $\mathbf{S}_{0,k}$ (SERMs) convert $a_{k,k}^{(k)}$ into 1s, where $s_k = (p_{k,k}^{(k)} - 1)/p_{k,N}^{(k)}$, and \mathbf{L}_k record the row multipliers used for the Gaussian elimination of column k . Then, we get

$$\begin{aligned} &\mathbf{L}_{N-1} \mathbf{P}_{N-1} \cdots \mathbf{L}_2 \mathbf{P}_2 \mathbf{L}_1 \mathbf{P}_1 \mathbf{A} \mathbf{S}_{0,1} \mathbf{S}_{0,2} \cdots \mathbf{S}_{0,N-1} \\ &= \begin{bmatrix} 1 & a_{1,2}^{(N-1)} & \cdots & a_{1,N}^{(N-1)} \\ 0 & 1 & \cdots & a_{2,N}^{(N-1)} \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & a_{N,N}^{(N-1)} \end{bmatrix} \\ &= \mathbf{D}_R \mathbf{U} \end{aligned} \quad (14)$$

where $a_{N,N}^{(N-1)} = e^{i\theta}$, $\mathbf{D}_R = \text{diag}(1, 1, \dots, 1, e^{i\theta})$, and

$$\mathbf{U} = \begin{bmatrix} 1 & a_{1,2}^{(N-1)} & \cdots & a_{1,N}^{(N-1)} \\ 0 & 1 & \cdots & a_{2,N}^{(N-1)} \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}. \quad (15)$$

Having, respectively, multiplied all the SERMs ($\mathbf{S}_{0,k}$) together, all the permutation matrices (\mathbf{P}_k) together, and all the unit lower triangular matrices (\mathbf{L}_k) together, we have one SERM \mathbf{S}_0^{-1} , one premultiplying permutation matrix \mathbf{P}^T , and one unit lower triangular matrix \mathbf{L}^{-1} . Since the inverse of a unit lower triangular matrix is also a unit lower triangular matrix and the transpose of a permutation matrix is also a permutation matrix, we have

$$\begin{aligned} &\mathbf{S}_{0,1} \mathbf{S}_{0,2} \cdots \mathbf{S}_{0,N-1} \\ &= \begin{bmatrix} 1 & & & \\ & \mathbf{I} & & \\ -s_1 & \cdots & -s_{N-1} & 1 \end{bmatrix} = \mathbf{S}_0^{-1} \\ &\mathbf{L}_{N-1} \mathbf{P}_{N-1} \cdots \mathbf{L}_2 \mathbf{P}_2 \mathbf{L}_1 \mathbf{P}_1 \\ &= \mathbf{L}_{N-1} (\mathbf{P}_{N-1} \mathbf{L}_{N-2} \mathbf{P}_{N-1}^T) \\ &\quad \cdots (\mathbf{P}_{N-1} \cdots \mathbf{P}_2 \mathbf{L}_1 \mathbf{P}_2^T \cdots \mathbf{P}_{N-1}^T) (\mathbf{P}_{N-1} \cdots \mathbf{P}_2 \mathbf{P}_1) \\ &= \mathbf{L}^{-1} \mathbf{P}^T \end{aligned} \quad (17)$$

where $\mathbf{L}^{-1} = \mathbf{L}_{N-1} (\mathbf{P}_{N-1} \mathbf{L}_{N-2} \mathbf{P}_{N-1}^T) \cdots (\mathbf{P}_{N-1} \cdots \mathbf{P}_2 \mathbf{L}_1 \mathbf{P}_2^T \cdots \mathbf{P}_{N-1}^T)$, $\mathbf{P}^T = \mathbf{P}_{N-1} \cdots \mathbf{P}_2 \mathbf{P}_1$.

Hence, we obtain $\mathbf{L}^{-1} \mathbf{P}^T \mathbf{A} \mathbf{S}_0^{-1} = \mathbf{D}_R \mathbf{U}$ or $\mathbf{A} = \mathbf{PLD}_R \mathbf{U} \mathbf{S}_0$.

If $\det \mathbf{A}$ is an integer factor, $a_{N,N}^{(N-1)} = e^{i\theta}$ is also an integer factor. Then, we can obtain its TERM factorization $\mathbf{A} = \mathbf{PLUS}_0$.

For a real matrix \mathbf{A} , if $\det \mathbf{A} = \pm 1$, then $e^{i\theta}$ is 1 or -1 , and the TERM factor matrices can also be real.

For a complex matrix \mathbf{A} , if $\det \mathbf{A} = e^{i\theta}$ is not an integer factor, then $a_{N,N}^{(N-1)} = e^{i\theta}$, which is a rotator for one complex number. The position of \mathbf{D}_R is flexible and can be extracted from and relocated between the TERM factor matrices, but then, the passed-by TERMS are all changed while the TERM properties are preserved. If $e^{i\theta}$ is an integer factor, the rotator can be multiplied by the unit upper TERM on its right or the unit lower TERM on its left in order to leave out the separate rotation matrix.

Having once gotten a TERM factorization, we can easily obtain the corresponding SERM factorization. (See proof of Theorem 4.)

V. COMPUTATIONAL STRUCTURES

In the previous two sections, we have proved the theorems and presented approaches for reversible integer mapping. For the N -by- N matrix \mathbf{A} , the number of factor TERMS is up to 3, and that of factor SERMs is up to $N+1$. A transform of either TERMS or SERMs can be sequentially computed in place.

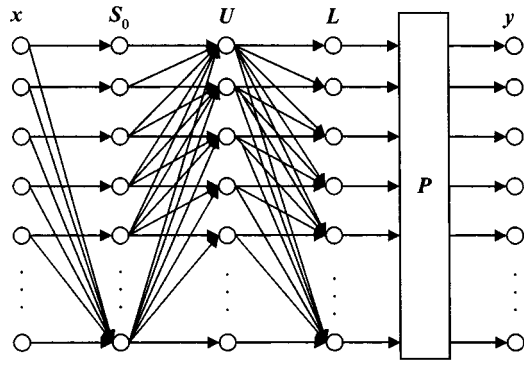


Fig. 2. Flowchart structure of the linear transform implemented by three TERMS.

A linear transform can be implemented by three TERMS with the flowchart structure of Fig. 2. The computation using the first TERM S_0 is only a transform to the N th element of the vector since this TERM is also a SERM. The sequential computational ordering of the upper TERM is from top to bottom and that of the lower TERM is from bottom to top. The computational ordering of its inverse is simply reversed.

The implementation with $N+1$ SERMs has a flowchart structure as illustrated in Fig. 3.

The inversion of TERM or SERM factorization can be implemented with a similar computing network by transferring data backward, subtracting instead of adding at the computing nodes, multiplying by the reciprocal of the integer factor, and simply reversing the sequential computational ordering. Both structures are composed of the elementary reversible structure—the ladder structure—shown in Fig. 1 and allows

- i) mapping integers to integers;
- ii) perfect reconstruction;
- iii) in-place calculation.

VI. COMPUTATIONAL COMPLEXITY AND OPTIMIZATION

Every transform built with TERMS or SERMs is immediately reversible, and its inverse transform has exactly the same computational complexity as the forward transform.

For the N -by- N matrix A , the computational complexity of TERM and SERM factorization and its original transform is listed in Table I.

The numbers in Table I appear in the extreme cases of linear transform matrices. Although the TERM and the SERM factorization structures are optimal, the factorization of a specific matrix can be practically optimized for lower computational complexity and less drift error by searching for a proper permutation matrix. An optimization scheme can be a minimization of

- i) the number of factorized matrices;
- ii) the computational complexity of each individual step;
- iii) the error with respect to the theoretical computation.

If the physical equivalents of a linear transform are ignored, the transform can be modified as a product of two unit TERMS. For $A = PLDU$, the modified transform matrix and the factorization can be $PD^{-1}P^T A = PD^{-1}LDU = PL_D U$, where $L_D = D^{-1}LD$ is a unit lower triangular matrix.

If the physical equivalents of a linear transform are taken into consideration, the relative magnitude of the transform

eigenvalues cannot be modified. Therefore, in order to obtain $|\det A| = 1$, matrix A can only be modified with a proportional scaling method. The factorization can then be optimized at lower computational expense for each individual TERM or SERM and with less accumulated error for the whole transform. Such optimization is not easy in practice, but the basic method should be complete pivoting, i.e., multiplying by both column and row permutation matrices $P_L^T A P_R^T = LUS_0 = S_N \cdots S_2 S_1 S_0$. The chosen pivots should bring forth the most zeros in the factor matrices and the most equal elements in a row, or as many as possible. This approach leads to the lower computational complexity of a transform.

To minimize the inherent error relative to the original theoretical computation, the permutations and the factorizations should lessen the error propagation and assist the error control. The error estimation method and some results are discussed in the next section.

However, the optimal factorization for a specific matrix still merits further investigation.

VII. ERROR ESTIMATION

An error is usually inevitable in floating-point number storage and calculation with fixed-precision computers. Since rounding arithmetic operations are applied in each ERM transform, there must be errors, compared with the original linear transform of theoretical or symbolic infinite-precision computation. If a rounding arithmetic is adopted to a complex number, the rounding-off errors of the real and the imaginary part are in the interval of $(-0.5, 0.5]$, and they will be in $[0, 1)$ if a chopping method is used. The intervals indicate a half-open rectangular region in the complex plane. For a TERM factorization of $A = PV_1 V_2 \cdots V_M$ with a denotation of \mathbf{u}_m for the rounding error vector that results from the transform of the m th TERM V_m and with $V_0 = P$, the total error vector of the whole transform is

$$\begin{aligned} \mathbf{u} &= V_0 \cdot (\mathbf{u}_1 + V_1 \cdot (\mathbf{u}_2 + V_2 \cdots (\mathbf{u}_{M-1} + V_{M-1} \cdot \mathbf{u}_M) \cdots)) \\ &= \sum_{m=1}^M \left(\prod_{k=0}^{m-1} V_k \right) \cdot \mathbf{u}_m. \end{aligned} \quad (18)$$

We use $\|\cdot\|_\infty$ for the infinity norm of vectors or matrices and u for the unit roundoff error, which is defined as the largest error that can occur in a rounding or chopping operation, i.e., 0.5 or 1 for real number transforms and $\sqrt{2}/2$ or $\sqrt{2}$ for complex number transforms. Then, an error bound can be estimated as

$$\begin{aligned} \|\mathbf{u}\|_\infty &= \left\| \sum_{m=1}^M \left(\prod_{k=0}^{m-1} V_k \right) \cdot \mathbf{u}_m \right\|_\infty \\ &\leq \sum_{m=1}^M \left\{ \left\| \prod_{k=0}^{m-1} V_k \right\|_\infty \cdot \|\mathbf{u}_m\|_\infty \right\} \\ &\leq u \cdot \sum_{m=1}^M \left\| \prod_{k=0}^{m-1} V_k \right\|_\infty. \end{aligned} \quad (19)$$

Splitting the complex matrix into an addition of a real and an imaginary matrix, the error bounds of the real and the imaginary part of the whole transform can analogously be given, respectively.

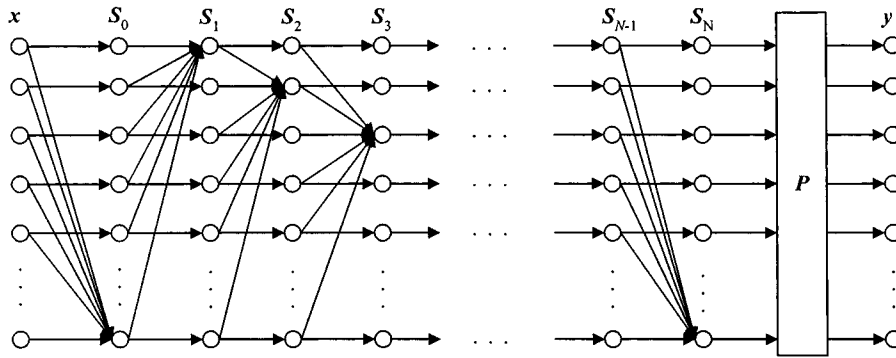


Fig. 3. Flowchart structure of the linear transform implemented by SERMs.

TABLE I
COMPUTATIONAL COMPLEXITY COMPARISON

Implementation	Addition	Multiplication	Rounding
Original transform	$N^2 - N$	N^2	N
TERM factorization	$N^2 - 1$	$N^2 - 1$	$2N - 1$
SERM factorization	$N^2 - 1$	$N^2 - 1$	$N + 1$

Apparently, if all the elements in a row of \mathbf{V}_m are integers, the corresponding element of \mathbf{u}_m will be 0. If all the elements of \mathbf{V}_m are integers, then $\mathbf{u}_m = \mathbf{0}$, and the matrix does not produce any fresh errors, but it still transfers and propagates existent errors that are produced by previous TERMS. Referring to the formulae of TERM and SERM factorization, all the elements in some rows are integers, and these rows bring in no primitive rounding errors. Thus, the accumulative error will never reach the upper bound of formula (19).

For the TERM factorization, the total rounding error is

$$\mathbf{u} = \mathbf{P} \cdot (\mathbf{u}_1 + \mathbf{L} \cdot (\mathbf{u}_2 + \mathbf{U}\mathbf{u}_3)) = \mathbf{P}(\mathbf{u}_1 + \mathbf{L}\mathbf{u}_2 + \mathbf{L}\mathbf{U}\mathbf{u}_3) \quad (20)$$

where only the N th element of \mathbf{u}_3 might not be 0. Therefore

$$|\mathbf{u}| = |\mathbf{P}(\mathbf{u}_1 + \mathbf{L}\mathbf{u}_2 + \mathbf{L}\mathbf{U}\mathbf{u}_3)| \\ \leq \mathbf{P}(|\mathbf{u}_1| + |\mathbf{L}\mathbf{u}_2| + |\mathbf{L}\mathbf{U}\mathbf{u}_3|)$$

and

$$\|\mathbf{u}\|_\infty \leq u \cdot (1 + \|\mathbf{L}\|_\infty + \|\mathbf{L}\mathbf{U}\mathbf{e}_N\|_\infty) \quad (21)$$

where $||$ denotes absolute values of all the elements in a vector.

For the SERM factorization, the total rounding error is

$$\mathbf{u} = \mathbf{P} \cdot (\mathbf{u}_1 + \mathbf{S}_N \cdot (\mathbf{u}_2 + \mathbf{S}_{N-1} \cdots (\mathbf{u}_N + \mathbf{S}_1 \cdot \mathbf{u}_{N+1}) \cdots)) \\ = \mathbf{P} \cdot \mathbf{u}_1 + \mathbf{P} \cdot \sum_{m=2}^{N+1} \left(\prod_{k=1}^{m-1} \mathbf{S}_{N-k+1} \right) \cdot \mathbf{u}_m \quad (22)$$

where only the $(N-m+1)$ th element of \mathbf{u}_m ($m = 1, 2, \dots, N$) and the N th element of \mathbf{u}_{N+1} might not be 0. Hence

$$|\mathbf{u}| \leq u \cdot \mathbf{P} \cdot \left\{ \sum_{m=N-1}^1 \left(\prod_{k=N}^{m+1} \mathbf{S}_k \right) \cdot \mathbf{e}_m \right. \\ \left. + \left(\mathbf{I} + \prod_{k=N}^1 \mathbf{S}_k \right) \cdot \mathbf{e}_N \right\}. \quad (23)$$

A factorization with partial or complete pivoting is necessary in order to minimize errors. However, the pivoting criterion still

merits further investigation since $p_{k,N}^{(k)} \neq 0$ mentioned in Section IV are not unique. We use $\min |s_m|$ for $m = k, \dots, N$ as a pivoting criterion in some of our experiments.

VIII. EXAMPLES

In order to demonstrate the availability and the practicability of the factorizations presented in this paper, the 4×4 discrete Fourier transform matrix in the complex number field, the eight-point discrete cosine transform matrix, and the Harr wavelet polyphase matrix in the real number field are used as examples for constructing the factorizations and to analyze the integer implementations.

A. Example 1: DFT

The 4×4 discrete Fourier transform matrix and its inverse are

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix} \\ \mathbf{A}^{-1} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{bmatrix}.$$

The matrix \mathbf{A} can reliably map integers to integers, but \mathbf{A}^{-1} cannot map integers to integers immediately and perfectly. Since $\det \mathbf{A} = 16i$, the transform matrix needs to be modified for the factorization.

If its physical equivalents are ignored, the transform can be modified and factorized into two unit TERMS. For $\mathbf{A} = \mathbf{P}\mathbf{L}\mathbf{D}\mathbf{U}$, $\mathbf{D} = \text{diag}(1, -1 - i, 2 - 2i, -4i)$, $\mathbf{P} = \mathbf{I}$, the modification and the factorization can be

$$\mathbf{P}\mathbf{D}^{-1}\mathbf{P}^T\mathbf{A} = \mathbf{D}^{-1}\mathbf{A} = \mathbf{L}\mathbf{U} \\ = \begin{bmatrix} 1 & & & \\ (-1+i)/2 & 1 & & \\ (1+i)/4 & (-1-i)/2 & 1 & \\ i/4 & (-1-i)/4 & (1-i)/2 & 1 \end{bmatrix} \\ \cdot \begin{bmatrix} 1 & 1 & 1 & 1 \\ & 1 & 1-i & -i \\ & & 1 & -i \\ & & & 1 \end{bmatrix}.$$

For the factorization $\mathbf{u}_2 = 0$, and such a modification produces $\mathbf{u} = \mathbf{u}_1$, in which the first component is 0. Then, $|\mathbf{u}| \leq u \cdot [0, 1, 1, 1]^T$.

The corresponding SERM factorization can be $PD^{-1}P^T \mathbf{A} = S_4 S_3 S_2 S_1$:

$$D^{-1} \mathbf{A} = LU = S_4 S_3 S_2 S_1 = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ -1/4 & (1-i)/4 & (1-i)/2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ & 1 & & \\ (-1+i)/4 & (-1-i)/2 & 1 & -i \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1 & 1 \\ (-1+i)/2 & 1 & 1-i & -i \\ & & 1 & \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1 & 1 \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix}.$$

Then, we can estimate the rounding error bounds $\mathbf{u}_4 = 0$ (which contribute 0 to the final error), $|\mathbf{u}_1| \leq u \cdot [0, 0, 0, 1]^T$, $|\mathbf{u}_2| \leq u \cdot [0, 0, 1, 0]^T$, $|\mathbf{u}_3| \leq u \cdot [0, 1, 0, 0]^T$, and we have $|\mathbf{u}| \leq u \cdot [0, 1, 1 + \sqrt{2}/2, 1 + 3\sqrt{2}/4]^T$.

If the dynamic ranges of all the elements of the original signal are $[0, L]$, then the dynamic ranges of the transformed data will be $[0, 4L]$, $[-L(1+i), L(1+i)]$ (a rectangular region in the complex plane), $[-2L, 2L]$, and $[-L(1+i), L(1+i)]$. If we require the dynamic ranges of the first and the third transformed components to be $[0, L]$ and $[-L/2, L/2]$, respectively, then the modification can be a multiplication of a diagonal matrix $D = \text{diag}(1/4, 1, 1/4, 1)$, which makes $\det(D\mathbf{A}) = \det D \det \mathbf{A} = i$ an integer factor. The factorization is somewhat more complicated.

$$D\mathbf{A} = PLUS_0$$

$$= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ & 1 & & \\ 1/4 & (-3-i)/8 & & 1 \\ 1/4 & (5+i)/8 & & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1/2 & -3/2 & 1 \\ & 1 & 1 & -2 \\ & & 1 & -1 \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ 0 & 1/2-i & 1/2 & i \end{bmatrix}$$

$$LU = S_4 S_3 S_2 S_1$$

$$= \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ -(3+i)/8 & (5+i)/8 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ & 1 & & \\ (5+i)/8 & -(3+i)/8 & 1 & -1 \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1/2 & -3/2 & 1 \\ 1 & 1 & 1 & -2 \\ & & 1 & \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1/2 & -3/2 & 1 \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix}.$$

When considering the physical equivalents, we must modify the transform by proportional scaling

$$\frac{1}{\sqrt[3]{16}} \mathbf{A} = \frac{1}{2} \mathbf{A} = PLUS_0 = \begin{bmatrix} 1 & & & \\ & 0 & 1 & \\ & 1 & 0 & \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ & 0 & & 1 \\ (1+i)/2 & (1-3i)/2 & & 1 \\ (1-i)/2 & (1+3i)/2 & -3 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1 & 1+2i & 1/2 \\ & 1 & -2i & -1/2 \\ & & 1 & -i/2 \\ & & & -i \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ -1 & 3 & -1-4i & 1 \end{bmatrix} \\ LU = S_4 S_3 S_2 S_1 = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ 2+i & 2-3i & -3 & -i \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ & 1 & & \\ (1+i)/2 & (1-3i)/2 & 1 & -i/2 \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ 0 & 1 & -2i & -1/2 \\ & & 1 & \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1 & 1+2i & 1/2 \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix}.$$

For TERM factorization $\mathbf{u}_3 = 0$, $|\mathbf{u}_2| \leq u \cdot [1, 1, 1, 0]^T$, $|\mathbf{u}_1| \leq u \cdot [0, 0, 1, 1]^T$, $\mathbf{u} = P\mathbf{u}_1 + PL\mathbf{u}_2$; then, we have $|\mathbf{u}| \leq u \cdot [1, 1 + \sqrt{5}, 1, 1 + \sqrt{5}]^T$.

The SERM factorization produces $\mathbf{u}_1 = \mathbf{u}_5 = 0$, which contribute nothing to \mathbf{u} . All but one element in each of $\mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_4$ are zeros, and these vectors satisfy $|\mathbf{u}_2| \leq u \cdot [0, 0, 1, 0]^T$, $|\mathbf{u}_3| \leq u \cdot [0, 1, 0, 0]^T$, and $|\mathbf{u}_4| \leq u \cdot [1, 0, 0, 0]^T$. Accordingly, the upper bound of the absolute value of the entire error vector is $u \cdot [1, 1 + (\sqrt{5} + 1)\sqrt{2}/2, 1, 3 + (\sqrt{5} + 1)\sqrt{2}/2]^T$.

Example 2: DCT

The 8×8 transform matrix of the eight-point discrete cosine transform (DCT) is shown in the first matrix at the bottom of the next page. The transform does not map integers to integers directly, but the matrix satisfies $\mathbf{A}^{-1} = \mathbf{A}^T$ and $\det \mathbf{A} = 1$. Therefore, it can be factorized into TERMS and SERMs without modification. In order to optimize the factorization,

we use a complete pivoting and a criterion for minimizing the magnitudes of the elements of \mathbf{S}_0 , and we find a factorization that results in the least errors: $\mathbf{P}_L^T \mathbf{A} \mathbf{P}_R^T = \mathbf{S}_8 \mathbf{S}_7 \cdots \mathbf{S}_2 \mathbf{S}_1 \mathbf{S}_0$, where

$$\mathbf{P}_L = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

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

$\mathbf{S}_m = \mathbf{I} + \mathbf{e}_m \mathbf{s}_m^T$, $m = 1, 2, \dots, 8$, and here, particularly, $\mathbf{S}_0 = \mathbf{I} - \mathbf{e}_8 \mathbf{s}_0^T$, $\mathbf{S}_0^{-1} = \mathbf{S}_0$. The SERM vectors are shown in the second matrix at the bottom of the page.

The upper bound of the rounding errors can be calculated from the above SERM factorization $|\mathbf{u}| \leq u \cdot [1.4619, 1.4842, 3.2816, 2.7699, 3.9169, 3.9778, 3.6444, 3.5648]^T$.

B. Example 3: DWT

The polyphase matrix of a discrete wavelet transform can first be factorized using Euclidean algorithm [9], after which there may be a remainder matrix of numbers. If the absolute value of the matrix determinant is 1, the matrix can be further factorized with the method proposed in this paper. The polyphase matrix of the analysis filters of unnormalized Harr wavelets is

$$\mathbf{P}(z) = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -1 & 1 \end{bmatrix} \quad \text{where } \det \mathbf{P}(z) = 1.$$

The TERM factorization and the SERM factorization of a 2×2 matrix are the same: at most three SERMs. Therefore, the factorization can be

$$\begin{aligned} \mathbf{P}(z) &= \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} -1 & 1 \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} \\ &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & 1 \end{bmatrix} \cdot \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix}. \end{aligned}$$

For comparison, the lifting scheme factorization by Daubechies and Sweldens [9] is copied as follows:

$$\mathbf{P}(z) = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}.$$

The most important difference between these two factorizations is that our algorithm allows integer factors on the diagonal.

$$\mathbf{A} = \begin{bmatrix} 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 \\ 0.4904 & 0.4157 & 0.2778 & 0.0975 & -0.0975 & -0.2778 & -0.4157 & -0.4904 \\ 0.4619 & 0.1913 & -0.1913 & -0.4619 & -0.4619 & -0.1913 & 0.1913 & 0.4619 \\ 0.4157 & -0.0975 & -0.4904 & -0.2778 & 0.2778 & 0.4904 & 0.0975 & -0.4157 \\ 0.3536 & -0.3536 & -0.3536 & 0.3536 & 0.3536 & -0.3536 & -0.3536 & 0.3536 \\ 0.2778 & -0.4904 & 0.0975 & 0.4157 & -0.4157 & -0.0975 & 0.4904 & -0.2778 \\ 0.1913 & -0.4619 & 0.4619 & -0.1913 & -0.1913 & 0.4619 & -0.4619 & 0.1913 \\ 0.0975 & -0.2778 & 0.4157 & -0.4904 & 0.4904 & -0.4157 & 0.2778 & -0.0975 \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{s}_0^T \\ \mathbf{s}_1^T \\ \mathbf{s}_2^T \\ \mathbf{s}_3^T \\ \mathbf{s}_4^T \\ \mathbf{s}_5^T \\ \mathbf{s}_6^T \\ \mathbf{s}_7^T \\ \mathbf{s}_8^T \end{bmatrix} = \begin{bmatrix} -1.1648 & 2.8234 & -0.5375 & 0.6058 & -1.2228 & 0.3805 & -0.0288 & 2 \\ 0 & -1.1129 & 0.0570 & -0.4712 & 0.1029 & 0.0156 & -0.4486 & -0.4619 \\ -0.0685 & 0 & 0.2708 & -0.2708 & -0.2235 & 0.2568 & -0.3205 & 0.3841 \\ -0.0364 & -1.7104 & 0 & -1.0000 & 0.3066 & 0.6671 & -0.5953 & 0.2039 \\ 0.7957 & 0.9664 & 0.4439 & 0 & 0.6173 & -0.1422 & 1.0378 & -0.1700 \\ 0.4591 & 0.4108 & -0.2073 & -1.0824 & 0 & 0.7071 & 0.8873 & -0.2517 \\ -0.6573 & 0.5810 & -0.2931 & -0.5307 & -0.8730 & 0 & -0.1594 & -0.3560 \\ 1.0024 & -0.7180 & -0.0928 & -0.0318 & 0.4170 & 1.1665 & 0 & 0.4904 \\ 1.1020 & -2.0306 & -0.3881 & 0.6561 & 1.2405 & 1.6577 & -1.1914 & 0 \end{bmatrix}$$

If we decompose a signal into two levels with Mallat's method, the matrix can be rewritten with four channels like

$$\begin{aligned} P(z) &= \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & & \\ -1 & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} \quad \text{where } \det P(z) = -1. \end{aligned}$$

It can be factorized as follows:

$$\begin{aligned} &\begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -\frac{1}{2} & -\frac{1}{2} & 1 \end{bmatrix} \\ &\cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & -1 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &\cdot \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \end{aligned}$$

This factorization shows that the calculation of the two-level DWT decomposition of a four-channel signal requires two shifts and seven adds, whereas the numbers are three shifts and six adds for the calculation of the two two-channel lifting steps. Moreover, our factorization need be calculated only once for both levels.

IX. CONCLUSIONS

- i) A linear transform has integer implementations (including a scaling modification if necessary), as long as it is invertible and finite dimensional.
- ii) Both TERM and SERM factorization can immediately be implemented by reversible integer mapping.
- iii) Both TERM and SERM implementations are perfectly invertible and in-place computable.
- iv) There are at most three TERMS or at most $N + 1$ SERMs for an $N \times N$ nonsingular matrix, except for a possible permutation matrix in the TERM or the SERM factorization. The numbers of ERMs are generally optimal.
- v) The factorization optimization of a specific matrix merits further investigation.

APPENDIX

This appendix contains the proof of Theorem 1, which has been presented in [11].

Its necessity is obvious, and a proof for its sufficiency is given as follows.

Lemma 2 shows that for a nonsingular matrix A , a triangular factorization of $A = PLDU$ exists. Then, we have $|\det D| = |\det PD| = |\det PLDU| = |\det A| = 1$. If we suppose $D = \text{diag}(d_1, d_2, \dots, d_N)$, $\lambda_m = d_1 d_2 \cdots d_m$ ($m = 1, 2, \dots, N$), $D_R = \text{diag}(1, \dots, 1, \lambda_N)$ and denote

$$D_O = \begin{cases} \text{diag}(\lambda_1, 1/\lambda_1, \lambda_3, 1/\lambda_3, \dots, \\ \lambda_{N-1}, 1/\lambda_{N-1}), & \text{if } N \text{ is even} \\ \text{diag}(\lambda_1, 1/\lambda_1, \lambda_3, 1/\lambda_3, \dots, \\ \lambda_{N-2}, 1/\lambda_{N-2}, 1), & \text{if } N \text{ is odd} \end{cases}$$

$$D_E = \begin{cases} \text{diag}(1, \lambda_2, 1/\lambda_2, \lambda_4, 1/\lambda_4, \dots, \\ \lambda_{N-2}, 1/\lambda_{N-2}, 1), & \text{if } N \text{ is even} \\ \text{diag}(1, \lambda_2, 1/\lambda_2, \lambda_4, 1/\lambda_4, \dots, \\ \lambda_{N-1}, 1/\lambda_{N-1}), & \text{if } N \text{ is odd} \end{cases}$$

then we obtain $D = D_O \cdot D_E \cdot D_R$ or $D = D_E \cdot D_O \cdot D_R$ and $\det D = \lambda_N = \det D_R = e^{i\theta}$, where D_R is a diagonal matrix: a rotation transform for the last complex component.

For a second-order matrix with two reciprocal diagonal elements, many forms of unit triangular factorization exist besides those given in the literature [9], such as

$$\begin{aligned} &\begin{bmatrix} \alpha & 0 \\ 0 & 1/\alpha \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 1/\alpha - 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \alpha - 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1/\alpha \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ -1/\alpha & 1 \end{bmatrix} \begin{bmatrix} 1 & \alpha - 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1/\alpha - 1 \\ 0 & 1 \end{bmatrix} \\ &= V_1 V_2 V_3 V_4. \end{aligned}$$

Therefore, with 2×2 partitioning skills, D_O and D_E can be decomposed as a product of four unit triangular matrices $D_O = V_{O1} V_{O2} V_{O3} V_{O4}$, $D_E = V_{E1} V_{E2} V_{E3} V_{E4}$ such that a complete factorization is

$$\begin{aligned} A = PLDU &= PLV_{O1} V_{O2} V_{O3} V_{O4} V_{E1} V_{E2} V_{E3} V_{E4} D_R U \\ &\text{or} \\ &= PLV_{E1} V_{E2} V_{E3} V_{E4} V_{O1} V_{O2} V_{O3} V_{O4} D_R U. \end{aligned}$$

The rotation matrix D_R can be relocated as the last term $D_R U = U_R D_R$, in which case, the two farthest left unit lower triangular matrices can be united as one unit lower triangular matrix V_1 , and the two farthest right unit upper triangular matrices can be united as one unit upper triangular matrix V_8 . The six middle matrices can be renamed as $V_2, V_3, V_4, V_5, V_6, V_7$. Consequently, the complete factorization of the transform has the form of

$$A = PV_1 V_2 V_3 V_4 V_5 V_6 V_7 V_8 D_R$$

which has the same structure as the one we expected to prove. The proof is completed. \blacksquare

Furthermore, the product of $V_4 \cdot V_5$ can be combined into one ERM if the factor elements are properly chosen. Therefore, M is not larger than 7.

ACKNOWLEDGMENT

The authors appreciate the hard work of Dr. W. Sun, Dr. T. Rogers, and Z. Liu in revising their manuscript and the great patience and the constructive comments of its anonymous reviewers.

REFERENCES

- [1] H. Blume and A. Fand, "Reversible and irreversible image data compression using the S-transform and Lempel-Ziv coding," *Proc. SPIE*, vol. 1091, pp. 2-18, 1989.
- [2] A. Zandi, J. Allen, E. Schwartz, and M. Boliek, "CREW: Compression with reversible embedded wavelets," in *Proc. IEEE Data Compression Conf.*, J. A. Storer and M. Cohn, Eds., Snowbird, UT, Mar. 1995, pp. 212-221.
- [3] A. Said and W. A. Pearlman, "An image multiresolution representation for lossless and lossy compression," *IEEE Trans. Image Processing*, vol. 5, pp. 1303-1310, Sept. 1996.
- [4] M. J. Gormish, E. L. Schwartz, A. F. Keith, M. P. Boliek, and A. Zandi, "Lossless and nearly lossless compression for high quality images," *Proc. SPIE*, vol. 3025, pp. 62-70, Mar. 1997.
- [5] F. A. M. L. Bruekers and A. W. M. van den Enden, "New networks for perfect inversion and perfect reconstruction," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 130-137, Jan. 1992.
- [6] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *J. Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186-200, 1996.
- [7] S. Dewitte and J. Cornelis, "Lossless integer wavelet transform," *IEEE Signal Processing Lett.*, vol. 4, pp. 158-160, June 1997.
- [8] A. R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo, "Wavelet transforms that map integers to integers," *J. Appl. Comput. Harmon. Anal.*, vol. 5, no. 3, pp. 332-369, 1998.
- [9] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 247-269, 1998.
- [10] Q. Shi, "Biorthogonal wavelet theory and techniques for image coding," *Proc. SPIE*, pp. 24-32, Oct. 1998.
- [11] P. Hao and Q. Shi, "Invertible linear transforms implemented by integer mapping" (in Chinese), *Sci. China*, ser. E, vol. 30, no. 2, pp. 132-141, 2000.
- [12] P. Hao, Q. Shi, and G. Xing, "Proposal of an integer reversible color transform," in *Proc. ISO/IEC JTC1/SC29/WG1N 1479*, Maui, HI, Dec. 6-10, 1999.

- [13] P. Hao and Q. Shi, "Proposal of reversible integer implementation for multiple component transforms," in *Proc. ISO/IEC JTC1/SC29/WG1N 1720*, Arles, France, July 3-7, 2000.
- [14] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1996.



Pengwei Hao (M'98) was born in the north of Shaanxi Province, China, in 1966. He received the B.Sc. degree in computer science and the M.Sc. degree in computer graphics from Northwestern Polytechnical University, Xi'an, China, in 1988 and 1994, respectively, and the Ph.D. degree in image processing from the Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, in 1997.

From 1997 to 1999, he was a Lecturer at the Center for Information Science, Peking University, Beijing.

In 2000, he was a Visiting Scientist for three months with the Centre for Vision, Speech, and Signal Processing, University of Surrey, Surrey, U.K. He is currently an Associate Professor with the Center for Information Science, Peking University. His research interests include data and image compression, data hiding, signal sampling and reconstruction, and computer graphics.



Qingyun Shi was born in Sichuan Province, China, in 1936. She graduated from the Mathematics Department, Peking University, Beijing, China, in 1957.

Since then, she has been teaching and doing research there in the field of image processing and analysis, pattern recognition, and applied mathematics. She is a Professor and the former Director of the National Laboratory on Machine Perception, Peking University. Her current research interests include visual information processing, image databases, and biometrics.

Prof. Shi is a member of the Chinese Academy of Sciences. From 1990 to 2000, she served as a member of the Governing Board of the International Association for Pattern Recognition.