Invertible Linear Transforms Implemented by Integer Mapping

(Published in Chinese, *Science in China, Series E*, April 2000, V30, N2, pp132-141)

HAO Pengwei SHI Qingyun
Center for Information Science, National Lab of Machine Perception
Peking University, Beijing, 100871

Abstract The general integer mapping theory of invertible linear transforms is considered in this paper. Two terms of integer factor and elementary triangular matrix are introduced. We prove that there exists an integer implementation if a linear transform is invertible and in finite dimensional space. The integer implementation of an invertible linear transform has several advantages such as (i) mapping integers to integers, (ii) perfectly invertible, and (iii) in-place calculation. A constructive approach of integer implementation and a simplified approach are also presented, which can be converted to some automatic steps. The error of integer implementation is estimated, and an example is given.

Keywords: Linear transforms, mapping integers to integers, triangular factorization, perfect inversion

1. Introduction

It is necessary for a finite word-length computer that a transform maps integers to integers and is perfectly invertible for hardware implementation and lossless compression of digital signals by means of transformation. A unified lossy/lossless compression system also enables region-of-interest (ROI) based image coding and progressive quality from the coarsest levels to the finer levels or even the lossless level. So integer transforms are significant for source coding.

Originally, people devised some integer transforms or some integer approaches of one-to-one mapping for some simple transforms, such as TS transform by Zandi et al (1995)[1] and their referred S transform by Blume & Fand(1989)[2], color space transform by Gormish et al (1997)[3]. In 1992, Bruekers & van den Enden[4] proposed a ladder structure of filtering networks in filter bank design for perfect inversion and perfect reconstruction, which broke a new path for integer transform research, but it was not paid enough attention to. To study integer transforms systematically with similar networks is for discrete wavelet transforms, in which the networks are utilized to construct wavelet and to lift the vanishing moment originally (Sweldens, 1996)[5], and the method is called the lifting scheme. A paper by Dewitte & Cornelis(1997)[6] seems to apply the lifting scheme to implement wavelet transforms by integer mapping firstly. Later, Daubechies & Sweldens et al (1998)[7,8] and Shi (1998)[9] extended and applied the lifting scheme to general wavelet transforms, and presented the schemes to convert discrete wavelet transform into integer transform steps, which were widely accepted and adopted.

The integer implementation of general linear transforms is addressed in this paper. The elementary structures of integer linear transforms are studied, and it is pointed out that ladder structures are the proper structures for integer implementation. Then, an approach to factorize an invertible linear transform into a series of elementary structures is presented, and an optimized factorization is proposed. However, to compare with the infinite precision symbolic calculation, an integer transform has errors resulted from arithmetic rounding operations, of which the upper bounds is also discussed. To lessen the errors, some principles are suggested. At the end of this paper, a typical example of 4-by-4 discrete Fourier transform is given to show the availability and practicability of the implementation.

2. Elementary structures of integer linear transforms

In following discussion, a linear transform is of complex number domain. An integer referred below in complex number field is considered as a complex number that both the real and imaginary components are integers. An arithmetic rounding operation for a complex number implies an integer conversion for both of its real and imaginary parts. Perfect inversion of a transform is suggested that both of the forward and the reverse transform are invertible.

A linear transform that maps number x to number y can be $y=ax+b$. If variable x can be assigned to an arbitrary integer, so the condition of integer output, y, is that both number a and number b are integers. There might be some processing procedures in the transform domain, so the range of y is considered as the set of integers without restrictions, and a condition for the inverse linear transform that maps integers to integers is that both $1/a$ and $1/b$ are integers. Therefore, the integer perfectly invertible condition of the linear transform is that $a=\pm 1$ or $\pm i$ and b is an integer, or the definition range of a or $1/a$ is $\{1, -1, i, -i\}$, where i is the imaginary unit. For recitative convenience, any element of the definition set is called an ***integer factor***, and denoted as j and $j'=1/j$. Obviously, j' is also an integer factor. If the variables are defined in Z transform domain, an integer factor may also be jz^k, where k is an integer.

If b is not an integer, the computations may be still done with floating point numbers, but an integer representation, $[b]$, can be a substitution, and the result is guaranteed to be integer and invertibility is preserved. The integer conversion methods can be rounding, chopping, carry-in, or some other styles.

It follows that a generic form of an invertible linear transform of a number implemented by integer mapping is: $y=jx+[b]$, and its inversion: $x=j'(y-[b])$. The flow chart is as follows:

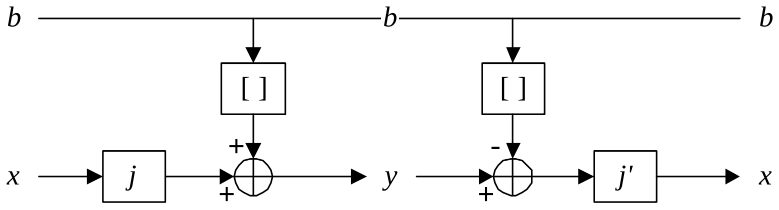

Fig. 1 Forward and reverse linear transform of a number implemented by integer mapping

As a matter of fact, the chart is also an elementary structure for the integer implementation. For invertible transforms, number b is necessary to satisfy two following conditions:

(i) For forward transform, b does NOT depend upon x directly or indirectly;

(ii) For reverse transform, b does NOT depend upon y directly or indirectly.

Whether in real or complex number field, the above structure is a kind of ladder structure. In the case of $a=1$, the structure is the same as the ladder structure proposed by Bruekers & van den Enden (1992)[4] and the lifting scheme structure by Sweldens (1996)[5]. So it can be summarized in brief that ladder structures are appropriate elementary structures for integer implementation of invertible linear transforms.

For a general N dimensional invertible linear transform $\boldsymbol{y}=\boldsymbol{A}\boldsymbol{x}$, if it allows perfectly invertible integer implementation, it must meet two necessary conditions of number b. It shows that if the calculational ordering is properly arranged there will be an elementary structure for perfectly invertible integer implementation, i.e. the transform matrix is an upper or lower triangular matrix whose diagonal elements are integer factors, which is called an ***elementary triangular matrix***. If all the diagonal elements of an elementary matrix are 1's, the matrix will be a unit triangular matrix.

An elementary triangular matrix has two important properties as follows:

(i) Product of two elementary upper triangular matrices is still an elementary upper triangular matrix, and product of two elementary lower triangular matrices is still an elementary lower triangular matrix.

(ii) Determinant of an elementary triangular matrix is an integer factor.

If ***A*** is an elementary upper triangular matrix, the calculational ordering of linear transform ***y***=***Ax*** can be arranged as top-down:

$$y_m = jx_m + \left[\sum_{n=m+1}^{N} a_{mn} x_n \right] = jx_m + [b] \qquad m = 1,2,3,\cdots,N \tag{1}$$

Its inverse ordering is also inverted:

$$x_m = j'\left(y_m - \left[\sum_{n=m+1}^{N} a_{mn} x_n \right] \right) = j'(y_m - [b]) \qquad m = N, N-1, N-2,\cdots,1 \tag{2}$$

If ***A*** is an elementary lower triangular matrix, the calculational ordering of linear transform ***y***=***Ax*** can be arranged as bottom-up, and its inversion top-down:

$$y_m = jx_m + \left[\sum_{n=1}^{m-1} a_{mn} x_n \right] = jx_m + [b] \qquad m = N, N-1, N-2,\cdots,1 \tag{3}$$

$$x_m = j'\left(y_m - \left[\sum_{n=1}^{m-1} a_{mn} x_n \right] \right) = j'(y_m - [b]) \qquad m = 1,2,3,\cdots,N \tag{4}$$

3. Factorization of invertible linear transforms

For arbitrary invertible linear transforms in finite dimension, whether there exists some integer implementation depends upon whether there exists a factorization consisting of a series elementary structures of integer implementation, or a series of elementary triangular matrices. For above linear transform, ***y***=***Ax***, the problem is whether there exists a factorization: $\boldsymbol{A} = \boldsymbol{V}_1\boldsymbol{V}_2\cdots\boldsymbol{V}_M$ or $\boldsymbol{A} = \boldsymbol{P}\boldsymbol{V}_1\boldsymbol{V}_2\cdots\boldsymbol{V}_M$, where ***P*** is a permutation matrix, $\boldsymbol{V}_i$ are elementary triangular matrices.

For arbitrary *N*-by-*N* invertible matrix ***A***, if $|\det \boldsymbol{A}| = a \neq 1$, the matrix can be modified to satisfy $|\det \boldsymbol{A}| = 1$. Without regard to the physical sense, the modification method can be a division by $m(\leq N)$ constants, $a_1, a_2, \cdots, a_m$, from *m* rows of ***A***. The constants satisfy:

$$\prod_{k=1}^{m} |a_k| = |\det \boldsymbol{A}| = a \tag{5}$$

An advantage of above modification is that the dynamic ranges of the transform components can be controlled, which is very helpful for the applications ignoring the physical meaning of a transform, such as lossless compression.

Take the physical implication into account, the relative magnitude of the transform eigenvalues cannot be modified, so a proportional modification is the alternative, or let $a_1 = a_2 = \cdots = a_N = \sqrt[N]{a}$.

Therefore, the case of $|\det \boldsymbol{A}| = 1$ is considered hereinafter. For $|\det \boldsymbol{A}| = 1$, following theorem can be derived:

Theorem: Matrix $\boldsymbol{A}$ has an elementary triangular factorization of $\boldsymbol{A} = \boldsymbol{P}\boldsymbol{V}_1\boldsymbol{V}_2\cdots\boldsymbol{V}_M\boldsymbol{D}_R$, if and only if $|\det \boldsymbol{A}| = 1$, where M is finite, $\boldsymbol{V}_k\,(k = 1,2,\cdots,M)$ are elementary triangular matrices, $\boldsymbol{P}$ is a permutation matrix, and $\boldsymbol{D}_R$ is a rotation matrix for one complex number.

Proof:

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

As deduced in matrix theory, for a nonsingular matrix $\boldsymbol{A}$, there exists a triangular factorization of $\boldsymbol{A}=\boldsymbol{PLDU}$, where $\boldsymbol{P}$, $\boldsymbol{L}$, $\boldsymbol{D}$, $\boldsymbol{U}$ are permutation matrix, unit lower triangular matrix, diagonal matrix and unit upper triangular matrix, respectively. $|\det \boldsymbol{D}| = |\det \boldsymbol{PD}| = |\det \boldsymbol{PLDU}\,| = |\det \boldsymbol{A}| = 1$. Suppose $\boldsymbol{D} = diag(d_1, d_2, \cdots, d_N)$, $l_n = d_1 d_2 \cdots d_n (n = 1,2,\cdots,N)$, $\boldsymbol{D}_R = diag(1,\cdots,1,\,l_N)$, and denote

$$\boldsymbol{D}_O = \begin{cases} diag(\,l_1, 1/l_1,\,l_3, 1/l_3, \cdots,\,l_{N-1}, 1/l_{N-1}) & \text{if } N \text{ is even} \\ diag(\,l_1, 1/l_1,\,l_3, 1/l_3, \cdots,\,l_{N-2}, 1/l_{N-2}, 1) & \text{if } N \text{ is odd} \end{cases}$$

$$\boldsymbol{D}_E = \begin{cases} diag(1,\,l_2, 1/l_2,\,l_4, 1/l_4, \cdots,\,l_{N-2}, 1/l_{N-2}, 1) & \text{if } N \text{ is even} \\ diag(1,\,l_2, 1/l_2,\,l_4, 1/l_4, \cdots,\,l_{N-1}, 1/l_{N-1}) & \text{if } N \text{ is odd} \end{cases}$$

then we obtain $\boldsymbol{D} = \boldsymbol{D}_O \cdot \boldsymbol{D}_E \cdot \boldsymbol{D}_R$ or $\boldsymbol{D} = \boldsymbol{D}_E \cdot \boldsymbol{D}_O \cdot \boldsymbol{D}_R$ and $\det \boldsymbol{D} = l_N = \det \boldsymbol{D}_R = e^{iq}$, where $\boldsymbol{D}_R$ is a diagonal matrix of rotation transform for the last complex component.

For a second-order matrix with two reciprocal diagonal elements, there exist many forms of unit triangular factorization besides those given in the literature [8], such as

$$\begin{aligned} \begin{bmatrix} a & 0 \\ 0 & 1/a \end{bmatrix} &= \begin{bmatrix} 1 & 0 \\ 1/a - 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ a-1 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1/a \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ -1/a & 1 \end{bmatrix} \begin{bmatrix} 1 & a-1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1/a - 1 \\ 0 & 1 \end{bmatrix} \\ &= \cdots \\ &= \boldsymbol{V}_1\boldsymbol{V}_2\boldsymbol{V}_3\boldsymbol{V}_4 \end{aligned} \tag{6}$$

Therefore, with the 2-by-2 partitioning skills, $\boldsymbol{D}_O$ and $\boldsymbol{D}_E$ can be decomposed as a product of 4 unit triangular matrices: $\boldsymbol{D}_O = \boldsymbol{V}_{O1}\boldsymbol{V}_{O2}\boldsymbol{V}_{O3}\boldsymbol{V}_{O4}$, $\boldsymbol{D}_E = \boldsymbol{V}_{E1}\boldsymbol{V}_{E2}\boldsymbol{V}_{E3}\boldsymbol{V}_{E4}$. Such that a complete factorization is:

$$\boldsymbol{A} = \boldsymbol{PLDU} = \boldsymbol{PLV}_{O1}\boldsymbol{V}_{O2}\boldsymbol{V}_{O3}\boldsymbol{V}_{O4}\boldsymbol{V}_{E1}\boldsymbol{V}_{E2}\boldsymbol{V}_{E3}\boldsymbol{V}_{E4}\boldsymbol{D}_R\boldsymbol{U} \tag{7}$$

or

$$= \boldsymbol{PLV}_{E1}\boldsymbol{V}_{E2}\boldsymbol{V}_{E3}\boldsymbol{V}_{E4}\boldsymbol{V}_{O1}\boldsymbol{V}_{O2}\boldsymbol{V}_{O3}\boldsymbol{V}_{O4}\boldsymbol{D}_R\boldsymbol{U} \tag{8}$$

The rotation matrix $\boldsymbol{D}_R$ can be relocated as the last term, $\boldsymbol{D}_R\boldsymbol{U} = \boldsymbol{U}_R\boldsymbol{D}_R$, then two farthest left unit lower triangular matrices can be united as one unit lower triangular matrix, $\boldsymbol{V}_1$, and two farthest right unit upper triangular matrices can be united as one unit upper triangular matrix $\boldsymbol{V}_8$. Six middle matrices can be renamed as $\boldsymbol{V}_2, \boldsymbol{V}_3, \boldsymbol{V}_4, \boldsymbol{V}_5, \boldsymbol{V}_6, \boldsymbol{V}_7$. So, the complete factorization of the transform has a form of

$$\boldsymbol{A} = \boldsymbol{P}\boldsymbol{V}_1\boldsymbol{V}_2\boldsymbol{V}_3\boldsymbol{V}_4\boldsymbol{V}_5\boldsymbol{V}_6\boldsymbol{V}_7\boldsymbol{V}_8\boldsymbol{D}_R \tag{9}$$

which has the same structure as expected to prove. The proof is completed. ■

The proven factorization exhibits at most 8 elementary integer transforms except a rotation and a permutation transform for an invertible linear transform. If e^{iq} is an integer factor, the rotation matrix can be multiplied by the last unit upper triangular matrix to make an elementary upper triangular matrix in order to left out the rotation transform. Without the rotation matrix, two corollaries can be deduced from the theorem:

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

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

If e^{iq} is not an integer factor, a complex rotation transform can be implemented with the real and the imaginary components. A rotation transform can be decomposed into 3 steps of elementary integer transforms, and a rotation matrix also has many unit triangular factorizations[8]:

$$\begin{aligned}\begin{bmatrix}\cos q & -\sin q\\ \sin q & \cos q\end{bmatrix} &= \begin{bmatrix}1 & 0\\ (1-\cos q)/\sin q & 1\end{bmatrix}\begin{bmatrix}1 & -\sin q\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ (1-\cos q)/\sin q & 1\end{bmatrix}\\ &= \begin{bmatrix}1 & (\cos q-1)/\sin q\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ \sin q & 1\end{bmatrix}\begin{bmatrix}1 & (\cos q-1)/\sin q\\ 0 & 1\end{bmatrix}\\ &= \cdots\\ &= \boldsymbol{V}_{R1}\boldsymbol{V}_{R2}\boldsymbol{V}_{R3}\end{aligned} \qquad (10)$$

Thus we proved that arbitrary finite dimensional invertible linear transform can be implemented by integer mapping or its matrix can be decomposed into a series of integer-invertible elementary triangular matrices if and only if the absolute value of the matrix determinant is 1. Thus, an arbitrary finite dimensional invertible linear transform has integer implementations, and every transform built with elementary triangular matrices is immediately invertible where the inverse transform has exactly the same computational complexity as the forward transform.

4. Implementation structures and optimization

In former section, we proved the theorem and presented an approach of integer implementation of an invertible linear transform. The number of elementary matrices is up to 8, the first and the last elementary triangular matrices are implemented in-place sequentially, and the middle 6 can be implemented in-place and parallel. Furthermore, it is easy to find that the position of $\boldsymbol{D}_R$ is flexible. It can be moved forward to any position in front of a unit triangular matrix, but then all the passed-by triangular matrices are changed while the unit triangularity is preserved.

In order to speed up the implementation, it is necessary to optimize the elementary triangular factorization. Optimization scheme can be (i) to minimize the number of factorized matrices, (ii) to minimize the computational complexity of each step.

If the physical sense of a linear transform is abandoned, the transform can be modified to be a product of two elementary unit triangular matrices. If ***A***=***PLDU***, factorization of the modified transform matrix will be $\boldsymbol{P}\boldsymbol{D}^{-1}\boldsymbol{P}^T\boldsymbol{A} = \boldsymbol{P}\boldsymbol{D}^{-1}\boldsymbol{L}\boldsymbol{D}\boldsymbol{U} = \boldsymbol{P}\boldsymbol{L}_D\boldsymbol{U}$, which can be implemented by integer mapping with two elementary integer transforms, where $\boldsymbol{L}_D = \boldsymbol{D}^{-1}\boldsymbol{L}\boldsymbol{D}$ is a unit lower triangular matrix.

If the physical significance of a linear transform is considered, the relative magnitude of the transform eigenvalues can not be modified, so the modification can only be with a scaling method and then to optimize the factorization. In fact, the factorization stated above can be combined. The product of $\boldsymbol{V}_4 \cdot \boldsymbol{V}_5$

with following elements can be:

$$\boldsymbol{V}_4\cdot\boldsymbol{V}_5=\begin{bmatrix}1 & -1/l_1 & & & & \\ & 1 & & & & \\ & & 1 & -1/l_3 & & \\ & & & 1 & & \\ & & & & 1 & \ddots \\ & & & & & \ddots\end{bmatrix}\cdot\begin{bmatrix}1 & & & & & \\ & 1 & & & & \\ & -1/l_2 & 1 & & & \\ & & & 1 & & \\ & & & -1/l_4 & 1 & \\ & & & & \ddots & \ddots\end{bmatrix}$$

$$=\begin{bmatrix}1 & -1/l_1 & & & & \\ & 1 & & & & \\ & -1/l_2 & 1 & -1/l_3 & & \\ & & & 1 & & \\ & & & -1/l_4 & 1 & \ddots \\ & & & & \ddots & \ddots\end{bmatrix} \tag{11}$$

The combined matrix can be calculated in-place and parallel. So, the whole transform can be implemented by integer mapping in 7 steps with a structure like:

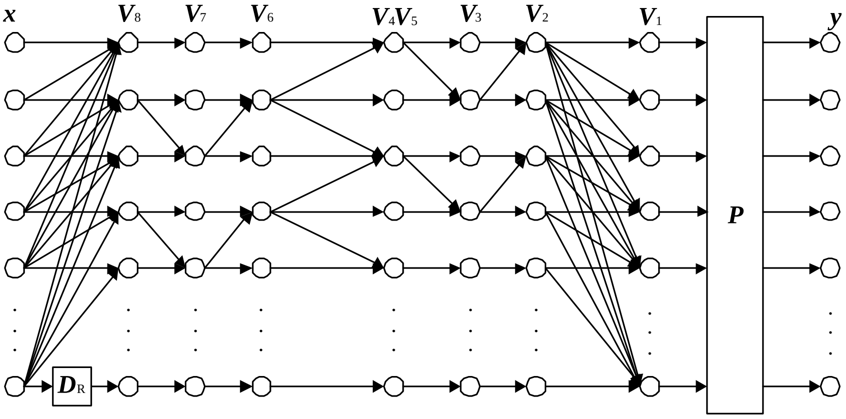

Fig. 2 A flowchart structure of linear transform implemented by integer mapping

The sequential calculating order of the first elementary triangular matrix is from top to bottom, and that of the last elementary triangular matrix is from bottom to top, while the middle 5 steps can be implemented with the inherent parallelism. The inversion can be implemented with similar computing network but transferring data backwards, subtracting instead of adding at the computing nodes and reversing the sequential calculating order. The structure allows the transform to be calculated in-place and without allocating auxiliary memory.

Elementary triangular factorization of a matrix can be obtained automatically by Gaussian elimination method. If a Gaussian elimination method is used with partial pivoting or complete pivoting, all the absolute values of the off-diagonal elements are not greater than 1 in the lower triangular matrix of LDU factorization, which is very helpful for the error control.

But undoubtedly, the forms of optimized factorization are not unique, and the optimal factorization still merits further investigation.

5. Error Estimation

Due to arithmetic rounding operations are applied in each elementary transform to make whole linear transform invertible by integer mapping, comparing to the original linear transform of infinite precision symbolic calculation, there must be errors. If a rounding-off arithmetic is adopted for a complex number, the rounding errors of the real and the imaginary part are in the interval of [-0.5,0.5), and they will be in that of [0,1) if a chopping method is used. The intervals are half-open rectangular region in complex plane. For a finite elementary factorization of $\boldsymbol{A} = \boldsymbol{P}\boldsymbol{V}_1\boldsymbol{V}_2\cdots\boldsymbol{V}_M$, with a denotation of $\boldsymbol{u}_m$ for the rounding error vector resulted from the transform of the *m*-th elementary matrix $\boldsymbol{V}_m$, and $\boldsymbol{V}_0 = \boldsymbol{P}$, then the total error vector of the whole transform is:

$$\boldsymbol{u} = \boldsymbol{V}_0 \cdot \left(\boldsymbol{u}_1 + \boldsymbol{V}_1 \cdot \left(\boldsymbol{u}_2 + \boldsymbol{V}_2 \cdots \left(\boldsymbol{u}_{M-1} + \boldsymbol{V}_{M-1} \cdot \boldsymbol{u}_M\right)\cdots\right)\right) = \sum_{m=1}^{M}\left(\prod_{k=0}^{m-1}\boldsymbol{V}_k\right)\cdot\boldsymbol{u}_m \qquad (12)$$

We use $\| \ \|_\infty$ for infinity norm of vectors or matrices and U for the upper bound of a unit rounding error. Since $\|\mathrm{Re}(\boldsymbol{u}_m)\|_\infty \le U$, $\|\mathrm{Im}(\boldsymbol{u}_m)\|_\infty \le U$ and $\|\boldsymbol{u}_m\|_\infty \le \sqrt{2}\cdot U$, an error bound can be estimated as:

$$\|\boldsymbol{u}\|_\infty = \left\|\sum_{m=1}^{M}\left(\prod_{k=0}^{m-1}\boldsymbol{V}_k\right)\cdot\boldsymbol{u}_m\right\|_\infty \le \sum_{m=1}^{M}\left\{\left\|\prod_{k=0}^{m-1}\boldsymbol{V}_k\right\|_\infty \cdot \|\boldsymbol{u}_m\|_\infty\right\} \le \sqrt{2}\cdot U\cdot\sum_{m=1}^{M}\left\|\prod_{k=0}^{m-1}\boldsymbol{V}_k\right\|_\infty \qquad (13)$$

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 whole transform can also be given analogously.

Apparently, if all the elements in a row of $\boldsymbol{V}_m$ are integers, the corresponding element of $\boldsymbol{u}_m$ will be 0. If all the elements of $\boldsymbol{V}_m$ are integers, $\boldsymbol{u}_m$ = 0, or the matrix does not bring in fresh errors but it still transfers and accumulates existent errors. Referring to the formulae (7,8,9) for elementary triangular factorization, there are at least 2 triangular matrix among $\boldsymbol{V}_1,\cdots,\boldsymbol{V}_8$ in which all the elements are integers, and they bring in no primitive rounding errors. Thus, the accumulative error will not reach the upper bound of formula (13) anyway.

The upper bound estimation shows: The further left a factor matrix is, the more its influence is on the final error. Therefore, optimized factorization should conform to following principles:

(i) Close to large numbers: In order not to amplify the progressive error, if there is an element absolutely larger than 1 in a factor matrix, the larger the number is, the nearer the matrix should be. The absolutely largest number had better appear in the last triangular matrix.

(ii) Away from large numbers: If there is an absolutely large number in a row of a matrix, all of the matrices on its left should avoid putting absolutely large numbers in the corresponding column. A better choice is the absolute value of all off-diagonal numbers in the corresponding column to be smaller than 1.

(iii) Prior for integers: An integer should be chosen if possible, so as not to round off error.

(iv) Factorizing with pivoting: A transform matrix should be factorized into elementary triangular matrices with partial or complete pivoting methods in order that all the off-diagonal elements of furthest left matrix are not larger than 1.

6. Examples

In order to show the availability and the practicability of the method presented in this paper, the 4-by-

4 discrete Fourier transform in complex number domain is considered to construct and to analyze the integer implementations.

The 4-by-4 discrete Fourier transform matrix and its inverse are:

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

The matrix $\boldsymbol{A}$ can map integers to integers without doubt, but $\boldsymbol{A}^{-1}$ can not map integers to integers immediately and perfectly. For det$\boldsymbol{A}$=16i, the integer implementation needs a modification for the transform matrix.

If the physical sense is ignored, the transform can be modified to be implemented with 2 elementary steps. Due to $\boldsymbol{A}=\boldsymbol{PLDU}$, $\boldsymbol{D}$=diag(1, -1-i, 2-2i, -4i), $\boldsymbol{P}=\boldsymbol{I}$, the modification and the decomposition can be:

$$\boldsymbol{PD}^{-1}\boldsymbol{P}^{T}\boldsymbol{A}=\begin{bmatrix}1 & & & \\ (-1+i)/2 & 1 & & \\ (1+i)/4 & (-1-i)/2 & 1 & \\ i/4 & (-1-i)/4 & (1-i)/2 & 1\end{bmatrix}\cdot\begin{bmatrix}1 & 1 & 1 & 1\\ & 1 & 1-i & -i\\ & & 1 & -i\\ & & & 1\end{bmatrix}$$

For the factorization, $\boldsymbol{u}_2=0$, and such modification makes $\boldsymbol{u}=\boldsymbol{u}_1$, in which the first component is 0.

If the dynamic ranges of all the elements of the original signal is [0, L], then the dynamic ranges of the transformed data will be [0, 4L], [-L(1+i), L(1+i)] (a rectangular region in complex plane), [-2L, 2L], and [-L(1+i), L(1+i)]. If we expect 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 with a diagonal matrix $\boldsymbol{D}$=diag(1/4, 1, 1/4, 1), which makes det($\boldsymbol{DA}$)=det$\boldsymbol{D}$ det$\boldsymbol{A}$= i being an integer factor. The factorization is a little bit more complicated:

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

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

The in-between diagonal matrix can be factorized into 4 unit triangular matrices using the standard factorization formula for order 2 diagonal matrix with two reciprocal elements:

$$\begin{bmatrix}1 & & & \\ & 2 & & \\ & & 1/2 & \\ & & & 1\end{bmatrix}=\begin{bmatrix}1 & & & \\ & 1 & & \\ & -1/2 & 1 & \\ & & & 1\end{bmatrix}\cdot\begin{bmatrix}1 & & & \\ & 1 & 1 & \\ & & 1 & \\ & & & 1\end{bmatrix}\cdot\begin{bmatrix}1 & & & \\ & 1 & & \\ & 1 & 1 & \\ & & & 1\end{bmatrix}\cdot\begin{bmatrix}1 & & & \\ & 1 & -1/2 & \\ & & 1 & \\ & & & 1\end{bmatrix}$$

Then combine the 2 furthest left lower matrices and the 2 furthest right upper matrices, we get:

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

Here, $\boldsymbol{u}_2=\boldsymbol{u}_3=0$, and they contribute 0 to the final error. Except that the second component of $\boldsymbol{u}_4$

might be ± 0.5, all the others are 0, which are added up to $\boldsymbol{u}$ as 0 or $[\pm(1-i)/8, 0, \pm(1+i)/8, \pm 1]^T$. Two former elements of $\boldsymbol{u}_1$ are 0, and two latter possibly non-zero elements are permuted to subscribe for the first and the third element of the final error $\boldsymbol{u}$. As a result, the upper bounds of the total error vector can be cumulated according to formula (12, 13) as $[1/8+U, 0, 1/8+U, 1]^T$ for absolute real part and $[1/8+U, 0, 1/8+U, 0]^T$ for absolute imaginary part.

Considering the physical significance, we have to modify the transform by scaling:

$$\frac{1}{\sqrt[4]{16}}\boldsymbol{A} = \frac{1}{2}\boldsymbol{A} = \begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ 1 & 1-i & 1 & \\ 1 & -i & -i & 1 \end{bmatrix} \cdot \begin{bmatrix} 1/2 & 1/2 & 1/2 & 1/2 \\ & (-1-i)/2 & -1 & (-1+i)/2 \\ & & 1-i & -1-i \\ & & & -2i \end{bmatrix}$$

$$= \begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ 1 & 1-i & 1 & \\ 1 & -i & -i & 1 \end{bmatrix} \cdot \begin{bmatrix} 1/2 & & & \\ & 1/(1-i) & & \\ & & 1-i & \\ & & & 2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1 & 1 \\ & -1 & i-1 & i \\ & & 1 & -i \\ & & & -i \end{bmatrix}$$

The in-between diagonal matrix can also be factorized into 4 unit triangular matrices using partitioning skills:

$$\begin{bmatrix} 1/2 & & & \\ & 1/(1-i) & & \\ & & 1-i & \\ & & & 2 \end{bmatrix} = \begin{bmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & -i & 1 & \\ 1 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 1 \\ & 1 & 1 & 0 \\ & & 1 & 0 \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & (i-1)/2 & 1 & \\ -1/2 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & -2 \\ & 1 & i-1 & 0 \\ & & 1 & 0 \\ & & & 1 \end{bmatrix}$$

The first lower triangular matrix can be combined with the lower matrix in former formula, and the last upper triangular matrix can be combined with the upper matrix in former formula, so the entire factorization can be of 4 elementary triangular matrices:

$$\frac{1}{2}\boldsymbol{A} = \begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ 1 & 1-2i & 1 & \\ 2 & -1-i & -i & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 1 \\ & 1 & 1 & 0 \\ & & 1 & 0 \\ & & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & (i-1)/2 & 1 & \\ -1/2 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1 & 1+2i \\ & -1 & 2i-2 & 1+2i \\ & & 1 & -i \\ & & & -i \end{bmatrix}$$

The factorization makes $\boldsymbol{u}_1 = \boldsymbol{u}_2 = \boldsymbol{u}_4 = 0$, which contribute nothing to $\boldsymbol{u}$. Two former elements of $\boldsymbol{u}_3$ are 0, and the real and the imaginary part of the third possibly non-zero element and the real part of the fourth element might be ± 0.5. Accordingly, the supremum of the absolute value of entire error vector is $[0.5, 1, 2.5, 2]^T$ for real part and $[0, 0.5, 0, 1.5]^T$ for imaginary part respectively.

7. Conclusions

(i) A linear transform has integer implementations as long as it is invertible and finite dimensional.

(ii) The integer implementation of a linear transform is perfectly invertible and in-place computable.

(iii) The steps of elementary integer transforms except a possible permutation operation is not more than 7 if the matrix determinant of the linear transform is an integer factor.

(iv) The optimization of the linear transform factorization merits further investigation.

References

[1] A. Zandi, J. Allen, E. Schwartz, and M. Boliek, CREW: Compression with reversible embedded wavelets, *IEEE Data Compression Conference*, Snowbird, UT, March 1995, pp. 212-221.

[2] H. Blume and A. Fand, Reversible and irreversible image data compression using the S-transform and Lempel-Ziv coding, *Proceedings of SPIE*, 1989, Vol. 1091(*Medical imaging III: Image Capture and Display*): 2-18.

[3] M. J. Gormish, E. L. Schwartz, A. F. Keith, M. P. Boliek, and A. Zandi, Lossless and nearly lossless compression for high quality images, *Proceedings of SPIE*, 1997, Vol. 3025(Very High Resolution and Quality Imaging II, March, 1997): 62-70

[4] F. A. M. L. Bruekers, A. W. M. van den Enden, New Networks for Perfect Inversion and Perfect Reconstruction, *IEEE J. on Selected Areas in Communications*, 1992, 10(1): 130-137.

[5] W. Sweldens, The Lifting Scheme: A Custom-Design Construction of Biorthogonal Wavelets, *J. of Applied and Computational Harmonic Analysis*, 1996, 3(2): 186-200.

[6] S. Dewitte, J. Cornelis, Lossless Integer Wavelet Transform, *IEEE Signal Processing Letters*, 1997, 4(6): 158-160.

[7] A. R. Calderbank, I. Daubechies, W. Sweldens, B-L. Yeo, Wavelet Transforms That Map Integers to Integers, *J. of Applied and Computational Harmonic Analysis*, 5(3): 332-369.

[8] I. Daubechies, W. Sweldens, Factoring Wavelet Transforms into Lifting Steps, *J. of Fourier Analysis and Application*, 1998, 4(3): 247-269.

[9] Shi Qingyun, Biorthogonal Wavelet Theory and Techniques for Image Coding, *Proceedings of SPIE*, 1998, Vol. 3545(ISMIP'98, Oct. 1998): 24-32.