IMPROVED INTEGER TRANSFORMS USING MULTI-DIMENSIONAL LIFTING

Ralf Geiger[1], *Yoshikazu Yokotani*[2], *Gerald Schuller*[1], *Jürgen Herre*[3]

[1]Fraunhofer IIS AEMT, Ilmenau, Germany
[2]University of Texas at Arlington, Electrical Engineering Dept., Arlington,TX, U.S.A.
[3]Fraunhofer IIS, Erlangen, Germany
Email: ggr@emt.iis.fhg.de, yoshi@msp.uta.edu, shl@emt.iis.fhg.de, hrr@iis.fhg.de

ABSTRACT

Recently lifting-based integer transforms have received much attention, especially in the area of lossless audio and image coding. The usual approach is to apply the lifting scheme to each Givens rotation. Especially in the case of long transform sizes in audio coding applications, this leads to a considerable approximation error in the frequency domain. This paper presents a multi-dimensional lifting approach for reducing this approximation error. In this approach, large parts of the transform are calculated without rounding operations, only the output is rounded and added. The new approach is applied and evaluated for both the Integer Modified Discrete Cosine Transform (IntMDCT) and the Integer Fast Fourier Transform (IntFFT).

1. INTRODUCTION

Usually integer transforms are obtained by decomposing the transform into Givens rotations and applying the lifting scheme [1] or ladder network [2, 3] to each Givens rotation [4, 5, 6]. This introduces a rounding error in each step. For succeeding stages of Givens rotations the rounding error accumulates. The resulting approximation error becomes a burden, especially for long transforms of e.g. 1024 spectral values, as used in audio coding applications. Specifically for the high frequency range, where audio signals usually contain a rather small amount of energy, the approximation error can be larger than the actual signal and becomes the main limiting factor for lossless coding efficiency.

So the main design objective for improved integer transforms is the reduction of the approximation error. Besides that, the computational complexity should also be considered. The current approach of applying the lifting scheme to each Givens rotation, including the trivial sum-difference-butterflies, considerably increases the computational complexity compared to the original non-integer version of the transform (by about a factor of 2).

Some publications focusing on lossless image coding [4], [7], [8] propose a reduction of the resulting approximation error by a generalized lifting decomposition. Unfortunately, this approach cannot simply be applied to long transforms used in audio coding because the resulting algorithm has a considerable computational complexity ($O(N^2)$) compared to fast, rotation-based algorithms ($O(N \log N)$).

The approach presented in [8] uses block matrices to recursively obtain a triangular matrix decomposition. While the approach presented here is based on similar block matrices, no recursion is necessary.

This paper is organized as follows: First the basic approach for multi-dimensional lifting is presented, then this approach is applied and evaluated both for the Integer Modified Discrete Cosine Transform (IntMDCT) and the Integer Fast Fourier Transform (IntFFT).

2. FROM CLASSICAL TO MULTI-DIMENSIONAL LIFTING

Usually the lifting scheme is applied to obtain an invertible integer approximation of a Givens rotation:

$$\begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} = \begin{pmatrix} 1 & \frac{\cos\alpha-1}{\sin\alpha} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \sin\alpha & 1 \end{pmatrix} \begin{pmatrix} 1 & \frac{\cos\alpha-1}{\sin\alpha} \\ 0 & 1 \end{pmatrix}$$

The integer approximation is achieved by applying a rounding function after each addition.

The lifting scheme can also be used for an invertible integer approximation of certain scaling operations. In [9] the following lifting decomposition of a 2×2 scaling matrix with determinant value of one is presented:

$$\begin{pmatrix} d & 0 \\ 0 & d^{-1} \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ d^{-1} & 1 \end{pmatrix} \begin{pmatrix} 1 & -d \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & d^{-1} \end{pmatrix}$$

This decomposition provides the basic idea for the new approach. The equation still holds when all the values are replaced by $n \times n$ matrices. Thus, for any invertible $n \times n$ matrix T and for the $n \times n$ identity matrix I_n the following decomposition of $2n \times 2n$ block matrices is possible:

$$\begin{pmatrix} T & 0 \\ 0 & T^{-1} \end{pmatrix} = \begin{pmatrix} -I_n & 0 \\ T^{-1} & I_n \end{pmatrix} \begin{pmatrix} I_n & -T \\ 0 & I_n \end{pmatrix} \begin{pmatrix} 0 & I_n \\ I_n & T^{-1} \end{pmatrix} \quad (1)$$

Apart from some simple operations, such as permutations or multiplication by -1, all the three blocks of this decomposition have the following general structure:

$$\begin{pmatrix} I_n & 0 \\ A & I_n \end{pmatrix}$$

with an $n \times n$ matrix A.

To this $2n \times 2n$ block matrix a generalized lifting scheme can be applied, called "multi-dimensional lifting" in this paper. Similar to the conventional lifting scheme, these $2n \times 2n$ matrices can be used for invertible integer approximations of the transform T in the following way: The first half of the integer input values are processed by the matrix A and then rounded to integer values before adding them to the second half of the values.

The inverse of the block matrix is given by

$$\begin{pmatrix} I_n & 0 \\ A & I_n \end{pmatrix}^{-1} = \begin{pmatrix} I_n & 0 \\ -A & I_n \end{pmatrix}.$$

In this way, the process can be inverted without any error by simply applying the same matrix A and the same rounding, and subtracting the resulting values instead of adding them. As the first half of the values is not modified in the forward step, they are still available for the inverse step. No special restrictions apply to the matrix A, e.g. it does not necessarily have to be invertible.

3. INTMDCT BY MULTI-DIMENSIONAL LIFTING

To obtain an invertible integer approximation of cosine modulated filter banks in general, they are decomposed into the windowing stage and the $\mathrm{DCT_{IV}}$, see [9]. For the MDCT the windowing stage becomes one stage of Givens rotations. Figure 1 illustrates this decomposition.

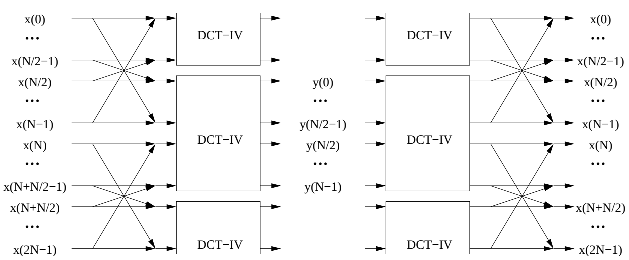

Fig. 1. Decomposition of MDCT and inverse MDCT into Givens rotations and $\mathrm{DCT_{IV}}$

Traditionally, the $\mathrm{DCT_{IV}}$ is decomposed into Givens rotations in the same way as for fast algorithms. The number of Givens rotations is given by $O(N \log N)$ for a transform of length N. The windowing stage in the MDCT decomposition consists of only $N/2$ Givens rotations or $3N/2$ rounding steps. Consequently, for transform lengths used in audio coding applications, e.g. 1024, the main contribution to the approximation error results from the integer approximation of the $\mathrm{DCT_{IV}}$ block.

The new approach presented in this paper utilizes the multi-dimensional lifting approach and in this way reduces the number of rounding steps in the $\mathrm{DCT_{IV}}$ to $3N/2$, which equals the number of rounding steps in the windowing stage (in contrast to $O(N \log N)$ rounding steps for the conventional lifting-based approach).

3.1. The Stereo IntMDCT

The most straight-forward way of using the multi-dimensional lifting approach for the IntMDCT is to apply the $\mathrm{DCT_{IV}}$ to two blocks of signals simultaneously. These blocks can either be from two subsequent blocks or from the left and the right channel of a stereo audio signal. The decomposition in equation (1) is applied to the $\mathrm{DCT_{IV}}$ matrix. Since the inverse of the $\mathrm{DCT_{IV}}$ is again the $\mathrm{DCT_{IV}}$, the decomposition in equation (1) becomes:

$$\begin{pmatrix} \mathrm{DCT_{IV}} & 0 \\ 0 & \mathrm{DCT_{IV}} \end{pmatrix} = \begin{pmatrix} -I_N & 0 \\ \mathrm{DCT_{IV}} & I_N \end{pmatrix} \begin{pmatrix} I_N & -\mathrm{DCT_{IV}} \\ 0 & I_N \end{pmatrix} \begin{pmatrix} 0 & I_N \\ I_N & \mathrm{DCT_{IV}} \end{pmatrix} \quad (2)$$

Thus, apart from permutations and multiplications with -1, the application of the $\mathrm{DCT_{IV}}$ to two blocks of signals can be performed with three multi-dimensional lifting steps. This process is illustrated in figure 2, including the rounding operations for the integer approximation.

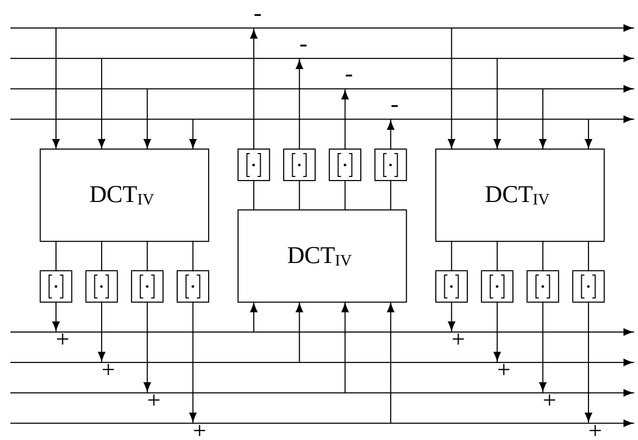

Fig. 2. Invertible integer approximation of two blocks of $\mathrm{DCT_{IV}}$ by three multi-dimensional lifting steps

With this approach, two $\mathrm{DCT_{IV}}$ transforms of length N can be implemented in an invertible integer fashion with only $3N$ rounding steps, i.e. $3N/2$ rounding steps per transform.

The $\mathrm{DCT_{IV}}$ in the three multi-dimensional lifting steps can have an arbitrary implementation, such as floating-point or fixed-point based, and does not need to be invertible. It just has to be performed in the same way in the forward and inverse IntMDCT. This makes the approach also suitable for high transform sizes. The overall computational complexity is about 1.5 times the computational complexity of the non-integer implementation of the two $\mathrm{DCT_{IV}}$ transforms. This is lower than for the conventional lifting-based integer implementations, which are about twice as complex as the conventional $\mathrm{DCT_{IV}}$, as these implementations have to implement the trivial $+/-$ butterflies based on lifting to achieve an energy conservation [6].

3.2. The Mono IntMDCT

The Stereo IntMDCT approach implies the simultaneous calculation of two $\mathrm{DCT_{IV}}$ transforms, e.g. by calculating the $\mathrm{DCT_{IV}}$ of two subsequent blocks or by calculating the $\mathrm{DCT_{IV}}$ of the left and the right channel simultaneously. While the first alternative introduces an additional delay of one block into the system, the second alternative is only possible for stereo signals. If neither the delay nor the stereo processing are desired, multi-dimensional lifting is still viable, but some additional stages of Givens rotations become necessary.

The $\mathrm{DCT_{IV}}$ of length N

$$\mathrm{DCT}_{\mathrm{IV}}^{(N)} = \left(\sqrt{\frac{2}{N}} \cos \frac{(2k+1)(2l+1)\pi}{4N} \right)_{k,l=0,\ldots,N-1}$$

can be decomposed into two $\mathrm{DCT_{IV}}$ of length $N/2$ and pre- and post-modulation stages. In the following, this decomposition is described.

Define the $N \times N$ matrices L and M by

$$\begin{pmatrix} L_{k,k} & L_{k,N-1-k} \\ L_{N-1-k,k} & L_{N-1-k,N-1-k} \end{pmatrix} =$$

$$\begin{pmatrix} \cos(\frac{2k+1}{4N}\pi) & -\sin(\frac{2k+1}{4N}\pi) \\ -\sin(\frac{2k+1}{4N}\pi) & -\cos(\frac{2k+1}{4N}\pi) \end{pmatrix} \quad k = 0, \ldots, N/2-1$$

$$L_{k,l} = 0 \qquad \text{else}$$

$$M = \frac{1}{\sqrt{2}} \begin{pmatrix} I_{N/2} & I_{N/2} \\ -I_{N/2} & I_{N/2} \end{pmatrix}$$

and the $N \times N$ permutation matrices P and Q by

$$P_{4k,4k} = P_{4k+1,4k+1} = P_{4k+2,4k+3} = P_{4k+3,4k+2} = 1$$
$$k = 0, \ldots, N/4-1$$
$$P_{k,l} = 0 \quad \text{else}$$

i.e. every second pair of values is swapped, and

$$Q_{k,2k} = Q_{N/2+k,2k+1} = 1 \quad k = 0, \ldots, N/2-1$$
$$Q_{k,l} = 0 \qquad \text{else}$$

i.e. the values with even indices are arranged first, and the values with odd indices are arranged subsequently.

With these matrices, the DCT_{IV} of length N can be decomposed into

$$\text{DCT}_{\text{IV}}^{(N)} = L \begin{pmatrix} \text{DCT}_{\text{IV}}^{(N/2)} & 0 \\ 0 & \text{DCT}_{\text{IV}}^{(N/2)} \end{pmatrix} MQP$$

Then the two DCT_{IV} of length $N/2$ can be decomposed into three multi-dimensional lifting steps of length $N/2$ using equation (2). The matrices L and M can both be considered as $N/2$ Givens rotations. The matrix M can be implemented using the multi-dimensional lifting steps

$$\begin{pmatrix} I_{N/2} & 0 \\ -\frac{1}{2}I_{N/2} & I_{N/2} \end{pmatrix} \begin{pmatrix} I_{N/2} & I_{N/2} \\ 0 & I_{N/2} \end{pmatrix}$$

The necessary scaling factors $\sqrt{2}$ and $1/\sqrt{2}$ can be handled by the DCT_{IV} stage. The lifting implementation of the matrices L and M can be combined with the DCT_{IV} stage to further reduce the overall number of rounding operations. This is done by merging the remaining $N/2$ rounding operations for M, and $N/2$ of the $3N/2$ rounding operations for L with the rounding operations in the DCT_{IV} stage [10]. So, overall only $5N/2$ rounding operations are necessary for this invertible integer approximation of the DCT_{IV} of length N.

Including the windowing stage, the total number of rounding operations for this IntMDCT is $4N$, i.e. 4 rounding operations per sample.

4. INTFFT BY MULTI-DIMENSIONAL LIFTING

The IntFFT [5] is an invertible integer approximation of the FFT. It can also be used to construct the Mono IntMDCT since a DCT_{IV} of length N can be further decomposed into a DFT of length $N/2$ with pre and post processing, all of which can be implemented by Givens rotations [11]. To apply the multi-dimensional lifting scheme for the IntFFT, the DFT of length N

$$\text{DFT}_N = \left(\frac{1}{\sqrt{N}} e^{\frac{j2\pi kl}{N}} \right)_{k,l=0,\ldots,N-1}$$

is first decomposed into the following decimation-in-frequency form:

$$\text{DFT}_N = \begin{pmatrix} \text{DFT}_{N/2} & 0 \\ 0 & \text{DFT}_{N/2} \end{pmatrix} RM, \tag{3}$$

where

$$R = \begin{pmatrix} I_{N/2} & 0 \\ 0 & \text{W}_{N/2} \end{pmatrix}, \quad M = \frac{1}{\sqrt{2}} \begin{pmatrix} I_{N/2} & I_{N/2} \\ I_{N/2} & -I_{N/2} \end{pmatrix},$$

$\text{DFT}_{N/2}$ is a DFT matrix length of $N/2$, and $\text{W}_{N/2}$ is an $N/2 \times N/2$ diagonal matrix whose diagonal elements are twiddle factors $e^{-j\frac{2\pi}{N}k}$ for $k = 0, \ldots, N/2-1$. Since it can be shown that a DFT can be realized by its inverse DFT (IDFT) by swapping the real and imaginary part as both pre- and post- processing [12], the first term of (3) will be

$$\begin{pmatrix} \text{DFT}_{N/2} & 0 \\ 0 & \text{DFT}_{N/2} \end{pmatrix} = J \begin{pmatrix} \text{DFT}_{N/2} & 0 \\ 0 & \text{DFT}_{N/2}^{-1} \end{pmatrix} J,$$

where

$$J = \begin{pmatrix} I_{N/2} & 0 \\ 0 & jI_{N/2}^* \end{pmatrix}.$$

I^* is a matrix so that all the elements of a matrix which are multiplied by I^* become the complex conjugate. The resulting form of the decomposition is

$$\text{DFT}_N = J \begin{pmatrix} \text{DFT}_{N/2} & 0 \\ 0 & \text{DFT}_{N/2}^{-1} \end{pmatrix} JRM \tag{4}$$

Now, the IntFFT of length N can be obtained by decomposing the second term of (4) into the following three multi-dimensional lifting steps of length $N/2$

$$\begin{pmatrix} \text{DFT}_{N/2} & 0 \\ 0 & \text{DFT}_{N/2}^{-1} \end{pmatrix} =$$

$$\begin{pmatrix} -I_{N/2} & 0 \\ \text{DFT}_{N/2}^{-1} & I_{N/2} \end{pmatrix} \begin{pmatrix} I_{N/2} & -\text{DFT}_{N/2} \\ 0 & I_{N/2} \end{pmatrix} \begin{pmatrix} 0 & I_{N/2} \\ I_{N/2} & \text{DFT}_{N/2}^{-1} \end{pmatrix}$$

and decomposing R and M into $3N/2 - 3$ and $3N$ classical lifting steps, respectively. It should be noted that the matrix R and the twiddle factors $\text{W}_{N/2}$ in the matrix R can be realized by the Givens rotations with rotational angles $\frac{\pi}{2}$ and $-\frac{2\pi}{N}k$ for $k = 0, \ldots, N/2-1$, respectively. The total number of rounding operations is $7.5N - 3$ which is about 3.75 rounding operations per real value. Note that a further reduction of rounding operations is possible by merging rounding operations, similar to the one described for the Mono IntMDCT.

5. RESULTS

The approximation accuracy of the multi-dimensional lifting based IntMDCT and IntFFT is evaluated by applying the transforms to the audio material used for the lossless audio coding activities of the ISO MPEG group [13]. The audio material consists of recordings of the New York Symphonic Ensemble and Jazz recordings. The evaluation is done for both 48 kHz / 16 bit and 96 kHz / 24 bit. The performance of the different transforms is evaluated by mean squared error (MSE), maximum absolute error and an entropy estimate, calculated by $\sum_k \log_2(2|y_k|+1)$, where y_k represents the integer spectral values. In the subsequent evaluation, the "number of instructions" value reflects the number of additions and multiplications.

5.1. IntMDCT

For the IntMDCT, a transform length of 1024 frequency bands is used. Table 1 shows the results for the conventional lifting based IntMDCT, the multi-dimensional lifting based Mono IntMDCT, and, as a reference, the rounded MDCT, which does not allow lossless operation. The resulting MSE and maximum absolute error values are similar for both audio input formats, thus only the overall values are displayed.

	Lifting based IntMDCT	Multi-dim. lifting IntMDCT	Rounded MDCT (not lossless)
Rounding operations per sample	22.5	4	1
Instructions per sample	45	32	20
MSE	1.97	0.48	0
max. abs. Error	8	4	0
Entropy estimate 48 kHz 16 bit	$1.180 \cdot 10^8$	$1.166 \cdot 10^8$	$1.160 \cdot 10^8$
Entropy estimate 96 kHz 24 bit	$4.145 \cdot 10^8$	$4.125 \cdot 10^8$	$4.113 \cdot 10^8$

Table 1. Comparison of conventional lifting-based IntMDCT, multi-dimensional lifting (MDL) based IntMDCT and rounded MDCT (not lossless)

It can be observed that the approximation error is largely reduced by the multi-dimensional lifting approach, and the estimated entropy comes close to the theoretical limit given by the rounded MDCT.

5.2. IntFFT

Similarly, the approximation accuracy of the multi-dimensional lifting based IntFFT is evaluated by comparing the number of rounding operations and instructions per real value, and MSE with those of the classical lifting based IntFFT and floating point FFT. The result is shown in Table 2. The size of the transforms is 512, which is the size necessary to implement an IntMDCT of length 1024.

	Lifting based IntFFT	Multi-dim. lifting IntFFT	Float FFT
Rounding operations per real value	16.67	3.75	0
Instructions per real value	33.34	24.02	15.00
MSE	2.69	0.70	0
max. abs. Error	8.24	3.16	0

Table 2. Comparison of conventional lifting-based IntFFT, multi-dimensional lifting (MDL) based IntFFT and float FFT

These results indicate that the multi-dimensional lifting based IntFFT can considerably reduce the number of rounding operations and the resulting MSE is reduced as well compared to the classical lifting based IntFFT. The computational complexity is about $3/2$ of that of the floating point FFT.

6. CONCLUSIONS

The proposed multi-dimensional lifting approach leads to a substantial reduction in rounding error compared to conventional one-dimensional lifting schemes for IntMDCT / IntFFT. Using this new approach, only 3 or 4 rounding steps per sample are required, independent of the transform length. This is particularly advantageous for large transform lengths (e.g. 1024) used in audio coding applications, compared to e.g. 22 rounding steps per sample for the conventional approach. The new approach decreases computational complexity significantly compared to previous approaches for integer transforms. More importantly, the reduction of rounding error results in an improved compression performance in the context of lossless audio coding applications.

7. REFERENCES

[1] I. Daubechies and W. Sweldens, "Factoring Wavelet Transforms into Lifting Steps," Tech. Rep., Bell Laboratories, Lucent Technologies, 1996.

[2] S. Mitra and R. Sherwood, "Digital ladder networks," *IEEE Transactions on Audio and Electroacoustics*, vol. AU-21, no. 1, pp. 30–36, February 1973.

[3] F. Bruekers and A. Enden, "New networks for perfect inversion and perfect reconstruction," *IEEE JSAC*, vol. 10, no. 1, pp. 130–137, Jan. 1992.

[4] K. Komatsu and K. Sezaki, "Reversible Discrete Cosine Transform," in *Proc. ICASSP*, 1998, vol. 3, pp. 1769–1772.

[5] S. Oraintara, Y.J. Chen, and T.Q. Nguyen, "Integer fast fourier transform," *IEEE Transactions on Signal Processing*, vol. 50, pp. 607–618, March 2002.

[6] R. Geiger, T. Sporer, J. Koller, and K. Brandenburg, "Audio Coding based on Integer Transforms," in *111th AES Convention*, New York, 2001.

[7] P. Hao and Q. Shi, "Matrix factorizations for reversible integer mapping," *IEEE Trans. Signal Processing*, vol. 49, pp. 2314 – 2324, October 2001.

[8] J. Wang, J. Sun, and S. Yu, "1-d and 2-d transforms from integers to integers," in *Proc. ICASSP'03*, Hong Kong, April 2003.

[9] R. Geiger and G. Schuller, "Integer low delay and MDCT filter banks," *Proc. of the Asilomar Conf. on Signals, Systems, and Computers*, 2002.

[10] R. Geiger, Y. Yokotani, and G. Schuller, "Improved integer transforms for lossless audio coding," in *Proc. of the Asilomar Conf. on Signals, Systems, and Computers*, 2003.

[11] Y. Yokotani and S. Oraintara, "Lossless audio compression using integer modified discrete cosine transform," in *Proc. ISPACS 2003*, Japan, December 2003.

[12] P. Duhamel, B. Piron, and J.M. Etcheto, "On computing the inverse DFT," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 285–286, February 1988.

[13] ISO/IEC JTC1/SC29/WG11 Moving Pictures Expert Group, "Final Call for Proposals on MPEG-4 Lossless Audio Coding," , no. N5208, October 2002.