A New Recursive Decomposition Algorithm for Realizing Fast IMDCT Computation

HUI LI, YIWEN WANG, PING LI
State key Laboratory of Electronic Thin Films and Integrated Devices
University of Electronic Science & Technology of China
Chengdu, Sichuan
CHINA
huili@uestc.edu.cn

Abstract: - In this paper, a new decomposition algorithm for realizing fast inverse modified discrete cosine transform (IMDCT) computation is presented. The decomposition algorithm can convert a pair of long transforms into 2 pairs of short transforms with half size of the long transforms. In addition, the decomposition algorithm can be utilized recursively to attain shorter transforms and reduce computational cycles. Based on the previous experimental results, the number of computational cycles of 512-point IMDCT is estimated to be 9641, which is less than those in previous reports.

Key-Words: inverse modified discrete cosine transform (IMDCT); DCT-IV; DST-IV; decomposition algorithm; TDAC; analysis/synthesis filter banks

1 Introduction

The modified discrete cosine transform (MDCT) and inverse modified discrete cosine transform (IMDCT) used to realize the analysis/synthesis filter banks of time domain aliasing cancellation scheme (TDAC) scheme [1,2] have been widely adopted in many applications [3–5]. Due to the intensive computational complexity of the MDCT and IMDCT, development of fast algorithms and simple architectures for implementing the MDCT/IMDCT computation becomes increasingly important, especially for realizing fast IMDCT computation in real-time audio decoding process.

Several fast algorithms and efficient architectures for implementing MDCT/IMDCT have been reported [6–20]. Cheng and Hsu [6] presented a systematic investigation of MDCT/IMDCT using a matrix representation. A radix-3 algorithm for efficiently computing the MDCT/IMDCT of a sequence with length $N = 2 \cdot 3^m$ was presented in [7]. Recently an efficient algorithm for implementing MDCT/IMDCT of transform block lengths $N = 5 \cdot 2^m$ (m≥2) was developed in [8]. To reduce computational complexity, the methods of decomposing MDCT/IMDCT into shorter DCT-IIIs or DCT-II/DST-II were presented [9–11]. Similarly, the MDCT/IMDCT is converted to shorter MDCTs/IMDCTs for realizing fast computation [7, 12].

Owing low implementation cost, the recursive algorithms and structures suitable for parallel VLSI implementation were thoroughly investigated [13–22]. Allowing for VLSI implementation, the algorithms presented in [19, 20] have the same computational efficiency whereas the architecture in [20] requires 1 latch less. Due to the lower computational latency of type-II inverse discrete cosine transform (IDCT-II) and the symmetry of IMDCT, the algorithm proposed in [21] possesses the computational efficiency 8 times of that in [19, 20] at the expense of approximately triple hardware cost. Compared with the algorithm in [21], our previously proposed decomposition algorithm of DCT-IV/DST-IV saves more than one third of computational cycles with smaller hardware implementation [22]. Absorbing virtues of the algorithm in [21] and [22], our recent proposed algorithm [23] possesses higher computational efficiency with smaller hardware accelerator compared with [22].

In this paper, a new decomposition algorithm for realizing fast inverse modified discrete cosine transform (IMDCT) computation is presented. The work presented in this paper is derived from the algorithm in [22]. Its aim is to further develop the algorithm in [22] and increase the computational efficiency for IMDCT computation. The core idea is to recursively decompose a pair of long transforms into 2 pairs of short transforms with half size of the long transforms. The detailed derivation of algorithm is proposed in section II. Conclusion is presented in Section III.
2 Decomposition Algorithm of IMDCT

2.1 Decomposing an N-point IMDCT into a pair of N/4-point transforms

The N-point IMDCT of an N/2-point input sequence \( x[k] \) is defined as follows: [2],[6]

\[
x[n] = \sum_{k=0}^{N/2-1} X[k] \cos \left( \frac{2n+1+\frac{k}{2}}{2N} \right) (2k+1) \pi,
\]

for \( n = 0, \ldots, N-1 \). (1)

The data sequence \( x[n] \) recovered by the backward MDCT is time-domain aliased data sequence which does not correspond to the original data sequence. It is assumed that the transform block length \( N \) equals to 16m, where \( m \) is a positive integer.

Defining another sequence \( y[n] \) as [9], [15]

\[
y[n] = \begin{cases} x \left[ n - \frac{N}{4} \right], & \text{for } n = \frac{N}{4}, \ldots, N-1, \\ -x \left[ n + \frac{3N}{4} \right], & \text{for } n = 0, \ldots, N-1, 
\end{cases}
\]

we can rewrite (1) as

\[
y[n] = \sum_{k=0}^{N/2-1} X[k] \cos \left( \frac{2n+1+\frac{k}{2}}{2N} \right) \frac{(2k+1)\pi}{4(N/2)} ,
\]

for \( n = 0, \ldots, N-1 \). (2)

The transform kernel in (3) is recognized as an N/2-point DCT-IV. The sequence of \( y[n] \) possesses the even antisymmetry property given by

\[
y[N-1-n] = \sum_{k=0}^{N/2-1} X[k] \cos \left[ \frac{2(N-1-n)+1+\frac{k}{2}}{2N} \right] \frac{(2k+1)\pi}{4(N/2)} \\
= -\sum_{k=0}^{N/2-1} X[k] \cos \left[ \frac{(2n+1+\frac{k}{2})\pi}{4(N/2)} \right] \\
= -y[n].
\]

(4)

That property allows computing only half of the coefficients in the sequence \( y[n] \), for \( n = 0, \ldots, N/2-1 \).

Let

\[
y[n] = \begin{cases} y[2n], & \text{for } n = 0, \ldots, N/4-1, \\
y[N-1-2n], & \text{for } n = N/4, \ldots, N-1, 
\end{cases}
\]

(5)

\[
y'[n] = \begin{cases} y[2n+1], & \text{for } n = 0, \ldots, N/4-1, \\
y[N-2-2n], & \text{for } n = N/4, \ldots, N-1. 
\end{cases}
\]

(6)

Equation (3) can be rewritten in 2 other forms:

\[
y[n] = \sum_{k=0}^{N/2-1} X[k] \cos \left( \frac{(4n+1)(2k+1)}{4N} \pi \right),
\]

for \( n = 0, \ldots, N/2-1 \) (7)

\[
y'[n] = \sum_{k=0}^{N/2-1} X[k] \cos \left( \frac{(4n+3)(2k+1)}{4N} \pi \right),
\]

for \( n = 0, \ldots, N/2-1 \). (8)

Based on (7) and (8), we can further derive

\[
z_{cc}[n]^{(0)} = \sum_{k=0}^{N/2-1} X[k] C(k) N \cos \left( \frac{(2n+1)(2k+1)}{N} \pi \right),
\]

(9)

\[
z_{cs}[n]^{(0)} = \sum_{k=0}^{N/2-1} X[k] S(k) N \sin \left( \frac{(2n+1)(2k+1)}{N} \pi \right),
\]

(10)

where

\[
\begin{align*}
z_{cc}[n]^{(0)} &= \frac{y'[n] + y[n]}{2}, & n = 0, \ldots, N/2-1. \\
z_{cs}[n]^{(0)} &= \frac{y'[n] - y[n]}{2}, & n = 0, \ldots, N/2-1. \\
C(k) &= \cos \left( \frac{(2k+1)}{2N} \pi \right), & k = 0, \ldots, N/2-1. \\
S(k) &= \sin \left( \frac{(2k+1)}{2N} \pi \right), & k = 0, \ldots, N/2-1.
\end{align*}
\]

(11)

From(5),(6), (11), we have

\[
y[2n] = z_{cc}[n]^{(0)} + z_{cs}[n]^{(0)}, & n = 0, \ldots, N/4-1,
\]

(12)

\[
y[2n+1] = z_{cc}[n]^{(0)} - z_{cs}[n]^{(0)}, & n = 0, \ldots, N/4-1.
\]

(13)

Therefore, the computation of an N-point IMDCT is converted to the computation of a pair of N/4-point transforms (\( z_{cc}[n]^{(0)} \) and \( z_{cs}[n]^{(0)} \)) instead.

2.2 Decomposing a pair of N/4-point transforms into two pairs of N/8-point transforms

The pair of N/4-point transforms presented in part A will be decomposed into two pairs of N/8-point transforms in this part. Then, the two pairs of N/8-point transforms will be recombined to get two new pairs of N/8-point transforms. And the relation between the transforms and \( y[n] \) is given. The derivation is as follows. Let

\[
Z_{cc}[k]^{(0)} = X[k] C(k),
\]

(14)

\[
Z_{cs}[k]^{(0)} = X[k] S(k).
\]

(15)

Substituting equations (14),(15) into (9),(10) separately, we have
\[ z_c(n)^{(0)} = \sum_{k=0}^{N/4-1} Z_c[k]^{(0)} \cdot \cos \left( \frac{(2n+1)(2k+1)}{N} \pi \right), \]  
(16)

\[ z_a(n)^{(0)} = \sum_{k=0}^{N/4-1} Z_a[k]^{(0)} \cdot \sin \left( \frac{(2n+1)(2k+1)}{N} \pi \right). \]  
(17)

Adopting the similar method of converting equation (3) to equations (9),(10),(12) and (13), from equation (16) we attain

\[ z_c[n]^{(1)} = \sum_{k=0}^{N/4-1} Z_c[k]^{(0)} C(k)_{N/4} \cdot \cos \left( \frac{(2n+1)(2k+1)}{N} \right) \pi, \]  
(18)

\[ z_a[n]^{(1)} = \sum_{k=0}^{N/4-1} Z_a[k]^{(0)} S(k)_{N/4} \cdot \sin \left( \frac{(2n+1)(2k+1)}{N} \right) \pi; \]  
(19)

\[ z_a[2n]^{(0)} = z_a[n]^{(0)} + z_a[n]^{(1)}, \quad \text{for } n = 0, \ldots, \frac{N}{8} - 1; \]  
(20)

and

\[ z_c[2n+1]^{(0)} = z_c[n]^{(0)} - z_c[n]^{(1)}, \quad \text{for } n = 0, \ldots, \frac{N}{8} - 1. \]  
(21)

From equation (16) we obtain

\[ z_c[n]^{(1)} = \sum_{k=0}^{N/4-1} Z_c[k]^{(0)} C(k)_{N/4} \cdot \cos \left( \frac{(2n+1)(2k+1)}{N} \right) \pi, \]  
(22)

\[ z_a[n]^{(1)} = \sum_{k=0}^{N/4-1} Z_a[k]^{(0)} S(k)_{N/4} \cdot \sin \left( \frac{(2n+1)(2k+1)}{N} \right) \pi, \]  
(23)

\[ z_a[2n]^{(0)} = z_a[n]^{(0)} + z_a[n]^{(1)}, \quad \text{for } n = 0, \ldots, \frac{N}{8} - 1, \]  
(24)

and

\[ z_c[2n+1]^{(0)} = z_c[n]^{(0)} - z_c[n]^{(1)}, \quad \text{for } n = 0, \ldots, \frac{N}{8} - 1. \]  
(25)

Based on the above derivation, computation of a pair of N/4-point transforms (equations (9) (10)) is converted to computation of two pairs of N/8-point transforms(equations (18)(19)(22)(23) . The relation between \( y[n] \) ( \( n = 0,1,\ldots,N/2-1 ) \) and \( z_a[n]^{(0)}, z_a[n]^{(1)}, z_c[n]^{(0)}, z_c[n]^{(1)} \) ( \( n = 0,1,\ldots,N/8-1 \) ) is given as follows.

Separating \( n \) in equations (12)(13) into odd and even numbers, we get

\[ y[4n] = z_c[2n]^{(0)} + z_c[2n]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(26)

\[ y[4n+2] = z_c[2n+1]^{(0)} + z_c[2n+1]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(27)

\[ y[4n+1] = z_c[2n]^{(0)} - z_c[2n]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(28)

and

\[ y[4n+3] = z_c[2n+1]^{(0)} - z_c[2n+1]^{(1)}, \quad n = 0, \ldots, N/8 - 1. \]  
(29)

Substituting equations (20)(21) (24)(25) into equations (26)(27)(28)(29), we have

\[ y[4n] = z_c[n]^{(0)} + z_a[n]^{(1)} + z_a[n]^{(0)} + z_a[n]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(30)

\[ y[4n+3] = ( z_c[n]^{(0)} + z_a[n]^{(1)} - (z_c[n]^{(0)} + z_a[n]^{(1)}), \quad n = 0, \ldots, N/8 - 1, \]  
(31)

\[ y[4n+1] = (z_c[n]^{(0)} - z_a[n]^{(1)} + (z_a[n]^{(0)} - z_a[n]^{(1)}), \quad n = 0, \ldots, N/8 - 1, \]  
(32)

\[ y[4n+2] = (z_c[n]^{(0)} - z_a[n]^{(1)} - (z_a[n]^{(0)} - z_a[n]^{(1)}), \quad n = 0, \ldots, N/8 - 1, \]  
(33)

Equations (30)(31)(32)(33) present the relation between \( \{y[n] \mid n = 0,1,\ldots,N/2-1\} \) and \( \{z_a[n]^{(0)}, z_a[n]^{(1)}, z_c[n]^{(0)}, z_c[n]^{(1)} \mid n = 0,1,\ldots,N/8-1 \} \).

To get two pairs of new N/8-point transforms, we need to recombine the sums in equations (30)(31)(32)(33). Let

\[ z_{cc}[n]^{(0)} = z_c[n]^{(0)} + z_c[n]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(34)

\[ z_{cc}[n]^{(1)} = z_c[n]^{(0)} + z_c[n]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(35)

\[ z_{cc}[n]^{(1)} = z_a[n]^{(1)} - z_{cc}[n]^{(0)}, \quad n = 0, \ldots, N/8 - 1, \]  
(36)

\[ z_{cc}[n]^{(0)} = z_a[n]^{(0)} - z_{cc}[n]^{(1)}, \quad n = 0, \ldots, N/8 - 1. \]  
(37)

Equations (30)(31)(32)(33) become

\[ y[4n] = z_{cc}[n]^{(0)} + z_{cc}[n]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(38)

\[ y[4n+3] = z_{cc}[n]^{(0)} - z_{cc}[n]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(39)

\[ y[4n+1] = z_{cc}[n]^{(0)} + z_{cc}[n]^{(1)}, \quad n = 0, \ldots, N/8 - 1, \]  
(40)

\[ y[4n+2] = z_{cc}[n]^{(0)} - z_{cc}[n]^{(1)}, \quad n = 0, \ldots, N/8 - 1. \]  
(41)
Substituing equations (18)(19)(22)(23) into equations (34)(35)(36)(37), we have

\[ z_n[n]^{(1)} = \sum_{k=0}^{N/2-1} Z_n[k]^{(0)} \cdot \cos \left( \frac{(2n+1)(2k+1)}{N} \pi \right) \]

\[ n = 0, \ldots, N/8 - 1, (42) \]

\[ z_n[n]^{(2)} = \sum_{k=0}^{N/2-1} Z_n[k]^{(0)} \cdot \sin \left( \frac{(2n+1)(2k+1)}{N} \pi \right) \]

\[ n = 0, \ldots, N/8 - 1, (43) \]

\[ z_n[n]^{(3)} = \sum_{k=0}^{N/2-1} Z_n[k]^{(0)} \cdot \cos \left( \frac{(2n+1)(2k+1)}{N} \pi \right) \]

\[ n = 0, \ldots, N/8 - 1, (44) \]

\[ z_n[n]^{(4)} = \sum_{k=0}^{N/2-1} Z_n[k]^{(0)} \cdot \sin \left( \frac{(2n+1)(2k+1)}{N} \pi \right) \]

\[ n = 0, \ldots, N/8 - 1 (45) \]

while

\[ Z_n[k]^{(1)} = Z_{cc}[k]^{(0)} C(k)_{N/2} - Z_{cs}[k]^{(0)} S(k)_{N/2} \]

\[ (46) \]

\[ Z_n[k]^{(2)} = Z_{cc}[k]^{(0)} S(k)_{N/2} + Z_{cs}[k]^{(0)} C(k)_{N/2} \]

\[ (47) \]

\[ Z_n[k]^{(3)} = Z_{cc}[k]^{(0)} C(k)_{N/2} + Z_{cs}[k]^{(0)} S(k)_{N/2} \]

\[ (48) \]

\[ Z_n[k]^{(4)} = Z_{cc}[k]^{(0)} S(k)_{N/2} - Z_{cs}[k]^{(0)} C(k)_{N/2} \]

\[ (49) \]

Equations (42)(43)(44)(45) are two pairs of new N/8-point transforms. From equations (30)(31)(32)(33) and equations (38)(39)(40)(41), we can see that \( y[n] \) is expressed as sum of 4 items using N/8-point transforms (equations (18)(19)(22)(23)) whereas sum of 2 items using new N/8-point transforms (equations (42)(43)(44)(45)). That is to say, using new N/8-point transforms can decrease computational complexity.

To clarify the relation among \( y[n] \) ( \( n = 0,1,\ldots,N/2-1 \)) and a pair of N/4-point transforms and two pairs of new N/8-point transforms, fig.1 is presented. The box of N/4-point SUM0 corresponds with equations (12) and the box of N/4-point SUB0 corresponds with equations (13). The four boxes in right column correspond with equations (38)-(41).

### 2.3 Conversion to DCT-IV/DST-IV

To reduce the computational cycles, equations (42) (43) (44) (45) are converted two pairs of N/8-point DCT-IV/DST-IV as

\[ z_n[n]^{(1)} = \sum_{k=0}^{N/8-1} Z_n[k]^{(2)} \cdot \cos \left( \frac{(2n+1)(2k+1)}{N} \pi \right) \]

\[ n = 0, \ldots, N/8 - 1 (50) \]

\[ z_n[n]^{(2)} = \sum_{k=0}^{N/8-1} Z_n[k]^{(2)} \cdot \sin \left( \frac{(2n+1)(2k+1)}{N} \pi \right) \]

\[ n = 0, \ldots, N/8 - 1 (51) \]

\[ z_n[n]^{(3)} = \sum_{k=0}^{N/8-1} Z_n[k]^{(2)} \cdot \cos \left( \frac{(2n+1)(2k+1)}{N} \pi \right) \]

\[ n = 0, \ldots, N/8 - 1 (52) \]

\[ z_n[n]^{(4)} = \sum_{k=0}^{N/8-1} Z_n[k]^{(2)} \cdot \sin \left( \frac{(2n+1)(2k+1)}{N} \pi \right) \]

\[ n = 0, \ldots, N/8 - 1 (53) \]

while

\[ Z_n[k]^{(2)} = Z_n[k]^{(0)} + Z_n\left[\frac{N}{2} - 1 - k\right]^{(0)} - Z_n\left[\frac{N}{4} - 1 - k\right]^{(0)} - Z_n\left[\frac{N}{4} + k\right]^{(0)} \]

\[ (54) \]

\[ Z_n[k]^{(3)} = Z_n[k]^{(0)} - Z_n\left[\frac{N}{2} - 1 - k\right]^{(0)} + Z_n\left[\frac{N}{4} - 1 - k\right]^{(0)} - Z_n\left[\frac{N}{4} + k\right]^{(0)} \]

\[ (55) \]

\[ Z_n[k]^{(4)} = Z_n[k]^{(0)} + Z_n\left[\frac{N}{2} - 1 - k\right]^{(0)} - Z_n\left[\frac{N}{4} - 1 - k\right]^{(0)} - Z_n\left[\frac{N}{4} + k\right]^{(0)} \]

\[ (56) \]

\[ Z_n[k]^{(5)} = Z_n[k]^{(0)} - Z_n\left[\frac{N}{2} - 1 - k\right]^{(0)} + Z_n\left[\frac{N}{4} - 1 - k\right]^{(0)} - Z_n\left[\frac{N}{4} + k\right]^{(0)} \]

\[ (57) \]
By using equations (46)-(49) (14)(15), $Z_{c}\{k\}^{m}$, $Z_{s}\{k\}^{s}$, $Z_{c}\{k\}^{1}$, $Z_{s}\{k\}^{1}$ in equations (54)-(57) can be expressed as functions of $X{k}$ in similar forms as ones in [22]. Equations (38)-(41), (50)-(53) have too similar correspondences in SPL. Therefore, there are similar computational cycles as [22] using equations (38)-(41), (50)-(53) to compute IMDCT. But the decomposition algorithm from a pair of N/4-point transforms to two pairs of N/8-point transforms in part B, section II can be used recursively to get four pairs of N/16-point transforms. In accordance, Equations (50)-(53) will become four pairs of N/16-point DCT-IV/DST-IV and computational cycles are reduced by 50%. Equations (54)-(57) will be in the form of eight items’ sum instead of four items’ sum and computational cycles are approximately doubled. Namely, the total computational cycles of 512-point IMDCT are approximately 9641 based on the experimental results of [22]. Therefore, the presented decomposition algorithm can reduce computational cycles of IMDCT compared with 12153 in [22] and 9718 in [23].

3 Conclusion
A new decomposition algorithm for realizing fast inverse modified discrete cosine transform (IMDCT) computation is presented in this paper. The decomposition algorithm can convert a pair of long transforms into 2 pairs of short transforms with half size of the long transforms. In addition, the decomposition algorithm can be utilized recursively to attain shorter transforms and reduce computational cycles. Based on the previous experimental results, the number of computational cycles of 512-point IMDCT is estimated to be 9641, which is less than 12153 in [22] and 9718 in [23]. The detailed number of computational cycles will be evaluated in future experiments.

Acknowledgment
The research work reported in this paper is supported by “the Fundamental Research Funds for the Central Universities”(Project Code: ZYGX2012J040).

References:


