e-ISSN: 2564-7954 CUSJE 22(2): 116-120 (2025) Research Article

Çankaya University

Journal of Science and Engineering

CUJSE

https://dergipark.org.tr/cankujse

Reduced Complexity Discrete Hilbert Transform Calculation for Finite Length Sequences

Orhan Gazi

[1] Department Electric Electronics Engineering, Ankara Medipol University, Ankara, Türkiye

Keywords

Discrete Hilbert Transform, Reduced Complexity, Matrix Decomposition.

Abstract

Hilbert transform is a well-known transform method used in communication systems. In this paper, we propose novel methods for the calculation of discrete Hilbert transform using matrix multiplication. The proposed method has reduced complexity, and it can be used for hardware implementations. Besides, we propose simple methods for the generation of Hilbert transform submatrices used in DHT operation. The proposed approaches can be used in hardware implementations.

1. Introduction

The Hilbert Transform is a matrix transformation, and it is used in signal processing and communication. It is used to shift the phase of a signal by 90 degrees and is employed to get an analytic signal. For a real-valued signal, the Hilbert transform can be used to get an imaginary part, turning the real-valued signal into a complex signal. Hilbert transform is used in communication systems, such as AM radio, medical imaging, such as MRI, ultrasound, and in signal modulation and demodulation.

The first papers on discrete Hilbert transform appeared in [1-5]. In [6] a discrete filtering technique based on circular convolution is presented. Generalization of fractional Hilbert transform is proposed in [7]. In [8] the authors propose the discrete fractional Hilbert transform and apply the proposed discrete fractional Hilbert transform to the edge detection of digital images. The authors in [9] propose a high-speed and reconfigurable discrete Hilbert transform architecture design methodology targeting real-time applications. The authors in [10] propose a reconfigurable architecture for the Hilbert transform to calculate any 4m-point (m is an integer) Hilbert transform. Performance and area are optimized based on the systolic-arrays method. In [11], the authors propose a novel configurable systolic array for discrete Hilbert transform (DHT) with any number of sample points. In [12], the authors propose a fast algorithm to compute the discrete Hilbert transform via the fast Hartley transform (FHT).

In [13] the authors proposed a low complexity no recursive fractional Hilbert transformer with small ripples and narrow transition using a merged of frequency response masking (FRM) and, frequency transformation (FT) approaches. An efficient method is shown by utilizing examples that the merging FRM-FT can achieve reduced computational complexity than earlier methods. In [14], the authors show a parallel implementation of Hilbert-Huang Transform on GPU focusing on reducing the computation complexity from O(N) on a single CPU to O(N/P log (N)) on GPU. In [15], the authors analyze the complexity of Hilbert transform filters (HTF) composed of two variable fractional delay filters rotated (VFDR). To reduce the complexity of designed filter the authors employ the symmetry of the coefficients representing the impulse response and ability to share the delay elements within the branch filters.

The continuous Hilbert transform of $x(t)$ is defined as

$$y(t) = x(t) * h(t) \tag{1}$$

* Corresponding Author: orhan.gazi@ankaramedipol.edu.tr
Received: September 9, 2025, Accepted: October 16, 2025

where

$$h(t) = \frac{1}{\pi t}$$

The convolution expression in (1) can be written as

$$y(t) = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x(\tau)}{t-\tau}d\tau$$

or as

$$y(t) = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x(t-\tau)}{\tau}d\tau$$

For digital periodic signal $x[n], x[n] = x[n+N]$, the DHT can be calculated as [1-3]

$$y[k] = \begin{cases} \frac{2}{N}\sum_{n,odd} x[n]\cot\frac{\pi}{N}(k-n), & k\ even \\ \frac{2}{N}\sum_{n,even} x[n]\cot\frac{\pi}{N}(k-n), & k\ odd \end{cases} \quad (2)$$

The inverse DHT formula is skew-symmetrical and it is given as [1-3]

$$x[n] = \begin{cases} -\frac{2}{N}\sum_{k,odd} y[k]\cot\frac{\pi}{N}(n-k), & n\ even \\ -\frac{2}{N}\sum_{k,even} y[k]\cot\frac{\pi}{N}(n-k), & n\ odd \end{cases} \quad (3)$$

where when $N \to \infty$, we have

$$\tan\frac{\pi n}{2N} \to \frac{\pi n}{2N} \qquad \cot\frac{\pi n}{2N} \to \frac{2N}{\pi n}$$

The discrete Hilbert transform for a finite length sequence is defined as [1-3]

$$y[k] = \begin{cases} \frac{2}{\pi}\sum_{n,odd}\frac{x[n]}{k-n} & k\ even \\ \frac{2}{\pi}\sum_{n,even}\frac{x[n]}{k-n} & k\ odd \end{cases} \quad (4)$$

The inverse discrete Hilbert transform for a finite length sequence is given as [1-3]

$$x[n] = \begin{cases} -\frac{2}{\pi}\sum_{k,odd}\frac{y[k]}{n-k} & n\ even \\ -\frac{2}{\pi}\sum_{k,even}\frac{y[k]}{n-k} & n\ odd \end{cases} \quad (5)$$

The matrix form of the DHT and inverse DHT can be written using matrices as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x}^{\boldsymbol{T}} \qquad \boldsymbol{x} = \boldsymbol{G}\boldsymbol{y}^{\boldsymbol{T}}$$

where $\boldsymbol{H}$ and $\boldsymbol{G}$ are the transform and inverse matrices. If the vector length of $\boldsymbol{x}$ is N, then the size of H and G is $N \times N$.

2. Reduced Complexity Discrete Hilbert Transform

It is important to have low complexity algorithms or mathematical expressions for efficient hardware implementations. Although significant improvements in electronic devices have been seen in the recent decades, still low complexity systems are preferable for hardware implementations.

The dimension of the $\boldsymbol{H}$ and $\boldsymbol{x^T}$ in

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x^T}$$

are $N \times N$ and N, and the matrix multiplication in

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x^T}$$

involves

$$N \times N \times N \to O(N^3)$$

multiplications and

$$N \times N \to O(N^2)$$

additions.

It is possible to decompose $H_{N\times N}$ matrix into two

$$\left(\frac{N}{2} \times \frac{N}{2}\right)$$

submatrices H_E andH_O.

Let $c_i, i = 1 \dots N$ and $r_j, j = 1 \dots N$ denote the column elements and row elements of the matrix H, the submatrix matrix H_E is obtained from even indexed column and odd indexed row elements of H, i.e.,

$$H_E = H(r_m, c_n) \quad m \ is \ odd, \ \ n \ is \ even$$

In a similar manner, the submatrix H_O can be obtained from $H_{N\times N}$ as

$$H_O = H(r_m, c_n) \quad m \ is \ even, \ \ n \ is \ odd$$

Then, the elements of $\boldsymbol{y}$ in $\boldsymbol{y} = \boldsymbol{Hx}$ can also be obtained using

$$\boldsymbol{y_E} = \boldsymbol{H_E x_O} \qquad \boldsymbol{y_O} = \boldsymbol{H_O x_E}$$

where $\boldsymbol{x_E}, \boldsymbol{y_E}$ and $\boldsymbol{x_O}.\boldsymbol{y_O}$ contain even indexed and odd indexed elements of $\boldsymbol{x}$ and $\boldsymbol{y}$.

The matrix multiplication in

$$\boldsymbol{y_E} = \boldsymbol{H_E x_O} \qquad \boldsymbol{y_O} = \boldsymbol{H_O x_E}$$

involves

$$2 \times \frac{N}{2} \times \frac{N}{2} \times \frac{N}{2} \to O\left(\left(\frac{N}{2}\right)^3\right)$$

multiplications and

$$2 \times \frac{N}{2} \times \frac{N}{2} \to O\left(\frac{N}{2}\right)$$

additions.

Example: For $N = 8$, $\boldsymbol{H_E}$ and $\boldsymbol{H_0}$ can be formed as

$$\boldsymbol{H_E} = \begin{bmatrix} -1 & -1/3 & -1/5 & -1/7 \\ 1 & -1 & -1/3 & -1/5 \\ 1/3 & 1 & -1 & -1/3 \\ 1/5 & 1/3 & 1 & -1 \end{bmatrix}$$

$$\boldsymbol{H_0} = \begin{bmatrix} 1 & -1 & -1/3 & -1/5 \\ 1/3 & 1 & -1 & -1/3 \\ 1/5 & 1/3 & 1 & -1 \\ 1/7 & 1/5 & 1/3 & 1 \end{bmatrix}$$

where it is seen that row and column dimensions of $\boldsymbol{H_E}$ and $\boldsymbol{H_0}$ are half of N.

Recursive Generation of $\boldsymbol{H_E}$

Let $r_i, i = 0 \ldots N/2 - 1$ denote the i^{th} row of $\boldsymbol{H_E}$**.** The$(i+1)^{th}$ row, r_{i+1}, of $\boldsymbol{H_E}$ can be obtained from r_i using

$$r_{i+1} = \left[\frac{1}{2i-1} \;\; r_i\left(1{:}\frac{N}{2}-1\right)\right]$$

where

$$r_i\left(1{:}\frac{N}{2}-1\right)$$

denotes the first $\left(\frac{N}{2}-1\right)$ elements of r_i.

Recursive Generation of $\boldsymbol{H_O}$

Let $r_i, i = 0 \ldots N/2 - 1$ denote the i^{th} row of $\boldsymbol{H_O}$**.** The$(i+1)^{th}$ row, r_{i+1}, of $\boldsymbol{H_O}$ can be obtained from r_i using

$$r_{i+1} = \left[\frac{1}{2i+1} \;\; r_i\left(1{:}\frac{N}{2}-1\right)\right]$$

where

$$r_i\left(1{:}\frac{N}{2}-1\right)$$

denotes the first $\left(\frac{N}{2}-1\right)$ elements of r_i.

3. Reduced Complexity Discrete Inverse Hilbert Transform

The inverse discrete Hilbert transform in (3) can be achieved using the matrix multiplication

$$\boldsymbol{x} = \boldsymbol{G}\boldsymbol{y}^{\boldsymbol{T}}$$

It is possible to decompose $G_{N\times N}$ matrix into two $\left(\frac{N}{2}\times\frac{N}{2}\right)$ submatrices G_E and G_O such that

$$G_E = G(r_m, c_n) \quad m \ \ is\ odd, \ \ n\ is\ even$$

$$G_O = G(r_m, c_n) \quad m \ \ is\ even, \ \ n\ is\ odd$$

Then, the elements of $\boldsymbol{x}$ in $\boldsymbol{x} = G\boldsymbol{y}$ can also be obtained using

$$\boldsymbol{x_E} = G_{\boldsymbol{E}}\boldsymbol{y_O} \qquad \boldsymbol{x_O} = \boldsymbol{G_O}\boldsymbol{y_E}$$

where $\boldsymbol{x_E}, \boldsymbol{y_E}$ and $\boldsymbol{x_O}.\boldsymbol{y_O}$ contain even indexed and odd indexed elements of $\boldsymbol{x}$ and $\boldsymbol{y}$.

The matrix multiplication in

$$\boldsymbol{x_E} = G_{\boldsymbol{E}}\boldsymbol{y_O} \qquad \boldsymbol{x_O} = \boldsymbol{G_O}\boldsymbol{y_E}$$

involves

$$2\times\frac{N}{2}\times\frac{N}{2}\times\frac{N}{2} \rightarrow O\left(\left(\frac{N}{2}\right)^3\right)$$

multiplications and

$$2\times\frac{N}{2}\times\frac{N}{2} \rightarrow O\left(\frac{N}{2}\right)$$

additions.

Declaration of Competing Interest

No conflict of interest was declared by the author(s).

Authorship Contribution Statement

Orhan Gazi: Methodology, Writing, Reviewing and Editing.

4. Conclusions

Hilbert transform is one of the basic transform methods used in communication engineering. In this paper, we introduced novel methods for the calculation of discrete Hilbert transform using matrix multiplications. The proposed method is low in complexity, and it can be employed hardware implementations. In addition, some simple methods are introduced for the generation of Hilbert transform submatrices, which are employed in DHT calculations. The proposed approaches can be used in hardware implementations.

References

[1] S. Kak, "The discrete Hilbert transform," Proc. IEEE, vol. 58, pp. 585-586, Apr. 1970, doi: 10.1109/PROC.1970.7696.

[2] V. Cizek, "Discrete Hilbert transform," IEEE Trans. Speech Audio Process, vol. 18, issue: 4, pp. 340 – 343, Dec. 1970, doi: 10.1109/TAU.1970.1162139.

[3] S. Kak, "Hilbert transformation for discrete data, " Int. J. Electron., pp. 177-183, 1973, doi: 10.1080/00207217308938428.

[4] M. Sabri, W. Steenaart, "Discrete Hilbert transform filtering, " IEEE Trans. Acoust., Speech, Signal. Process., vol. 25, issue: 5, pp. 452 – 454, Oct.1977, doi: 10.1109/ICASSP.1976.1170068.

[5] R. Ansari, "IIR discrete-time Hilbert transformers," IEEE Trans., ASSP-33, pp. 1146-1150, Aug.1985, doi: 10.1109/TASSP.1987.1165250.

[6] S. L. Hahn, "Hilbert Transforms," Transforms and Applications Handbook, A. D. Poularikas, Ed., CRC Press Inc., Boca Raton, FL, 1996.

[7] R. Tao, X. Li, Y. Wang, "Generalization of the fractional Hilbert transform," IEEE Signal Process. Lett., vol. 15, pp. 365 – 368, Dec. 2008, doi: 10.1109/LSP.2008.919814.

[8] S.C. Pei, M.H. Yeh, "Discrete fractional Hilbert transform," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process, vol. 47, issue: 11, Dec.2000, doi: 10.1109/82.885138.

[9] P. S. Reddy, S. Mopuri, A. Acharyya, "A Reconfigurable high speed architecture design for discrete Hilbert transform," IEEE Signal Process. Lett., vol. 21, issue. 11, pp. 1413 – 1417, Nov. 2014, doi: 10.1109/LSP.2014.2333745.

[10] L. Liu, Y. Zhang, "Design and Implementation of Reconfigurable Discrete Hilbert Transform Based on Systolic-Arrays," presented at the CMC 2010, Shenzhen, China, Apr. 12-14, 2010, pp.245-249.

[11] R. Sun, Y. Zhang, J. Chen, "The configurable structure for discrete Hilbert Transform via systolic array," presented at the ICIS 2025 (IEEE), Niigata, Japan, Jun. 16-20, 2013, pp. 467-473.

[12] S.C. Pei, S.B. Jaw, "Computation of discrete Hilbert transform through fast Hartley transform," IEEE Trans. Circuits Syst. , vol. 36, issue. 9, pp. 1251 – 1252, Sept. 1989, doi: 10.1109/31.34675.

[13] M.Pushpalatha, "Design of Low Complexity Nonrecursive Fractional Hilbert Transformer," presented at the ICMACC 2022, Hyderabad, India, Dec. 28-30, 2022, pp.1-5.

[14] P. Waskito, S. Miwa, Y. Mitsukura, H, Nakajo, "Parallelizing Hilbert-Huang Transform on a GPU," presented at the ICNDC 2010, Higashi, Japan, Nov. 17-19, 2010, pp. 184-190.

[15] E. Hermanowicz, A. Paruzel, "Farrow structure for complex digital Hilbert filter of low complexity," presented at the SPA 2007, Poznan, Poland, Sept. 7-7, 2007, pp.91-96.