**International Journal of Electrical and Computer Engineering (IJECE)**
Vol. 13, No. 2, April 2023, pp. 1533~1540
ISSN: 2088-8708, DOI: 10.11591/ijece.v13i2.pp1533-1540


# New fast Walsh–Hadamard–Hartley transform algorithm

**Suha Suliman Mardan, Mounir Taha Hamood**
Electrical Engineering Department, College of Engineering, University of Tikrit, Tiktrit, Iraq

**Article Info**

***Article history:***

Received Apr 28, 2022
Revised Sep 20, 2022
Accepted Oct 15, 2022

***Keywords:***

Discrete Hartley transform
Fast algorithms
Kronecker product
Orthogonal transforms
Walsh–Hadamard transform

**ABSTRACT**

This paper presents an efficient fast Walsh–Hadamard–Hartley transform (FWHT) algorithm that incorporates the computation of the Walsh-Hadamard transform (WHT) with the discrete Hartley transform (DHT) into an orthogonal, unitary single fast transform possesses the block diagonal structure. The proposed algorithm is implemented in an integrated butterfly structure utilizing the sparse matrices factorization approach and the Kronecker (tensor) product technique, which proved a valuable and fast tool for developing and analyzing the proposed algorithm. The proposed approach was distinguished by ease of implementation and reduced computational complexity compared to previous algorithms, which were based on the concatenation of WHT and FHT by saving up to 3N-4 of real multiplication and 7.5N-10 of real addition.



***Corresponding Author:***

Mounir Taha Hamood
Electrical Engineering Department, College of Engineering, University of Tikrit
P O BOX 42, Tikrit, Iraq
Email: m.t.hamood@tu.edu.iq

## 1. INTRODUCTION

The orthogonal transforms and their fast algorithms have a vital role in a variety of fields such as signal processing [1], image encryption [2], [3] digital watermarking [4], wireless communication systems [5]–[7] and many other fields [8]–[10]. The importance of these transforms has prompted numerous researchers to use various techniques and methodologies, which have developed a wide variety of fast algorithms and novel transforms to solve current problems and challenges or satisfy the criteria of modern applications. One of the most effective approaches to creating new transforms is combining two orthogonal transforms to provide a realistic and cost-effective solution while retaining quality by benefiting from their unique advantages, besides exchanging resources [11]–[15].

Discrete Hartley transform (DHT) is a significant member of the orthogonal transformations family, representing an efficient tool for a wide range of applications [2], [4], [15]–[18]. The importance of DHT is due to its use as an alternative to the discrete Fourier transform (DFT) for real input; therefore, utilizing DHT achieves a significant increase in computing efficiency by saving arithmetic operations and memory storage. Additionally, the DHT has the self-inverse characteristic, which means that except for the scale factor, the same algorithm is used for forward and inverse [16], [17], [19]. Another essential member of orthogonal transforms is the Walsh-Hadamard transform (WHT), a common tool in a wide range of applications [3], [5], [8], [20]–[23]. WHT's key distinguishing feature is that its computation does not include any multiplication or division.

Therefore, in this paper, a radix-2 fast Walsh–Hadamard–Hartley transform (FWHT) algorithm integrates the computation of both WHT and DHT into a single fast algorithm for the length power-of-two sequences. The primary advantage of the radix-2 FWHT algorithm is that it concurrently computes both transforms (WHT and FHT) utilizing a single butterfly. Furthermore, the FWHT has more efficient performance and lower arithmetic complexity than the traditional technique based on the concatenation of fast WHT and DHT algorithms. Accordingly, the proposed algorithm can be applied to a wide range of applications, such as in orthogonal frequency division multiplexing (OFDM) systems.

The remainder of the paper is structured in the following manner. The development of the proposed algorithm is completely derived in section 2. Section 3 discusses the applications of the developed algorithm. Section 4 discusses computational complexity and comparisons. The conclusion is presented in section 5.

## 2. DERIVATION OF THE ALGORITHM

Radix-2 FWHT starts by constructing the $H_N$ matrix, which is equal to the product of the WHT and DHT matrices as (1):

$$H_N = \frac{1}{N} WH_N \hat{\mathrm{DH}}_N \tag{1}$$

where $H_N$ is used to denote the H-transform matrix, $WH_N$ is the Walsh–Hadamard matrix, and $\hat{\mathrm{DH}}_N$ denotes the discrete Hartley by rearranging the rows in bit reversed order. The DHT matrix in (1) has order $N$ and may be expressed in terms of the lower order $N$/2. [24], [25] as shown in (2).

$$\hat{\mathrm{DH}}_N = \begin{bmatrix} \hat{\mathrm{DH}}_{\frac{N}{2}} & \hat{\mathrm{DH}}_{\frac{N}{2}} \\ \hat{\mathrm{DH}}_{\frac{N}{2}} D_{\frac{N}{2}} & -\hat{\mathrm{DH}}_{\frac{N}{2}} D_{\frac{N}{2}} \end{bmatrix} \tag{2}$$

$$\mathrm{D}_{N/2} = \mathrm{diag}(\cos(\tfrac{2\pi k}{N})) + \mathrm{diag}(\sin(\tfrac{2\pi k}{N})) \times (1 \oplus \mathrm{J}_{\frac{N}{2}-1}) \tag{3}$$

where $0 \le k \le \frac{N}{2} - 1$ and $J_N$ is the exchange matrix of order $N$ (i.e., reverse diagonal unity matrix) specified by:

$$\mathrm{J}_N = \begin{bmatrix} & & & 1 \\ & & 1 & \\ & \iddots & & \\ 1 & & & \end{bmatrix}$$

as shown in (2) may be factorized into the (4).

$$\begin{aligned} \hat{\mathrm{DH}}_N &= \begin{bmatrix} \hat{\mathrm{DH}}_{N/2} & 0 \\ 0 & \hat{\mathrm{DH}}_{N/2} \end{bmatrix} \begin{bmatrix} \mathrm{I}_{N/2} & \mathrm{I}_{N/2} \\ \mathrm{D}_{N/2} & -\mathrm{D}_{N/2} \end{bmatrix} \\ &= \begin{bmatrix} \hat{\mathrm{DH}}_{N/2} & 0 \\ 0 & \hat{\mathrm{DH}}_{N/2} \end{bmatrix} \begin{bmatrix} \mathrm{I}_{N/2} & 0 \\ 0 & \mathrm{D}_{N/2} \end{bmatrix} \begin{bmatrix} \mathrm{I}_{N/2} & \mathrm{I}_{N/2} \\ \mathrm{I}_{N/2} & -\mathrm{I}_{N/2} \end{bmatrix} \end{aligned} \tag{4}$$

The factorization shown in (4) can be stated in terms of tensor (Kronecker) product as (5).

$$\hat{\mathrm{DH}}_N = \left(\mathrm{I}_2 \otimes \hat{\mathrm{DH}}_{N/2}\right) \Delta_N \;\; \mathrm{DH}_2 \otimes \mathrm{I}_{N/2} \tag{5}$$

Using similar approach, $\mathrm{WH}_N$ can be written as (6):

$$\mathrm{WH}_N = \mathrm{I}_2 \otimes \mathrm{WH}_{N/2} \quad \mathrm{WH}_2 \otimes \mathrm{I}_{N/2} \tag{6}$$

Substituting (5) and (6) into (1), the general radix-2 H-matrix of order $N$=2$^{m}$ can be expressed as (7).

$$H_N = \frac{1}{N}\ I_2 \otimes WH_{2^{m-1}}\quad WH_2 \otimes I_{2^{m-1}}\ \left(I_2 \otimes \hat{DH}_{2^{m-1}}\right)\Delta_{2^m}\ \ DH_2 \otimes I_{2^{m-1}} \tag{7}$$

The terms product $WH_2 \otimes I_{2^{m-1}}\ \left(I_2 \otimes \hat{DH}_{2^{m-1}}\right)$ can be exchanged with each other, with the aid of a result of the multiplication rule for tensor product [26].

Therefore (7) can be written as (8):

$$\begin{aligned} H_N &= \frac{1}{N}\ I_2 \otimes WH_{2^{m-1}}\ \left(I_2 \otimes \hat{DH}_{2^{m-1}}\right)\ WH_2 \otimes I_{2^{m-1}}\quad \Delta_{2^m}\ \ DH_2 \otimes I_{2^{m-1}} \\ &= \frac{1}{N}\left(I_2 \otimes WH_{2^{m-1}}\ \hat{DH}_{2^{m-1}}\right)\ WH_2 \otimes I_{2^{m-1}}\quad \Delta_{2^m}\ \ DH_2 \otimes I_{2^{m-1}} \\ &= \frac{1}{N}\left(I_2 \otimes WH_{2^{m-1}}\ \hat{DH}_{2^{m-1}}\right) H_N^{I} \end{aligned} \tag{8}$$

$$H_N^{I} = \ WH_2 \otimes I_{2^{m-1}}\quad \Delta_{2^m}\ \ DH_2 \otimes I_{2^{m-1}} \tag{9}$$

with the aid of (6) and (5), respectively, the product $WH_{2^{m-1}}\ \hat{DH}_{2^{m-1}}$ can be factorized to,

$$\hat{DH}_{2^{m-1}} = \left(I_2 \otimes \hat{DH}_{2^{m-2}}\right)\Delta_{2^{m-1}}\ \ DH_2 \otimes I_{2^{m-2}} \tag{10}$$

$$WH_{2^{m-1}} = \ I_2 \otimes WH_{2^{m-2}}\quad WH_2 \otimes I_{2^{m-2}} \tag{11}$$

Substituting $WH_{2^{m-1}}$ and $\hat{DH}_{2^{m-1}}$ by their values in (10) and (11) into (8), we obtain,

$$H_N = \frac{1}{N}\left(I_2 \otimes\ I_2 \otimes WH_{2^{m-2}}\quad WH_2 \otimes I_{2^{m-2}}\ \left(I_2 \otimes \hat{DH}_{2^{m-2}}\right)\Delta_{2^{m-1}}\ \ DH_2 \otimes I_{2^{m-2}}\ \right) H_N^{I} \tag{12}$$

Employing the same strategy as in (7); therefore, (12) can be expressed as (13):

$$H_N = \frac{1}{N}\left(I_4 \otimes WH_{2^{m-2}}\ \hat{DH}_{2^{m-2}}\right) H_N^{II}\ H_N^{I} \tag{13}$$

where

$$H_N^{II} = \ I_2 \otimes\ WH_2 \otimes I_{2^{m-2}}\quad I_2 \otimes \Delta_{2^{m-1}}\quad I_2 \otimes DH_2 \otimes I_{2^{m-2}} \tag{14}$$

This factorization will be repeated, and after $\log_2 N$ stages, the final stage will be denoted as (15):

$$\begin{aligned} \left(I_{2^{m-2}} \otimes WH_4\ \hat{DH}_4\right) &= \ I_{2^{m-2}} \otimes\ \ I_2 \otimes WH_2\quad WH_2 \otimes I_2\quad I_2 \otimes DH_2\ \ \Delta_4\ \ DH_2 \otimes I_2 \\ &= \ I_{2^{m-2}} \otimes\ \ I_2 \otimes WH_2\quad I_2 \otimes DH_2\quad WH_2 \otimes I_2\ \ \Delta_4\ \ DH_2 \otimes I_2 \\ &= \ I_{2^{m-1}} \otimes WH_2\quad I_{2^{m-1}} \otimes DH_2\quad I_{2^{m-2}} \otimes WH_2 \otimes I_2\quad I_{2^{m-2}} \otimes \Delta_4\quad I_{2^{m-2}} \otimes DH_2 \otimes I_2 \end{aligned} \tag{15}$$

By combining (9)–(15) and utilizing the fact $\Delta_2 = I_2$ ,The H-matrix can be decomposed into,

$$H_N = \frac{1}{N}\; I_{2^{m-1}} \otimes WH_2 \quad I_{2^{m-1}} \otimes \Delta_2 \quad I_{2^{m-1}} \otimes DH_2 \quad I_{2^{m-2}} \otimes WH_2 \otimes I_2 \quad I_{2^{m-2}} \otimes \Delta_4$$
$$\times\; I_{2^{m-2}} \otimes DH_2 \otimes I_2 \;\cdots\cdots\; I_2 \otimes WH_2 \otimes I_{2^{m-2}} \quad I_2 \otimes \Delta_{2^{m-1}} \quad I_2 \otimes DH_2 \otimes I_{2^{m-2}} \tag{16}$$
$$\times\; WH_2 \otimes I_{2^{m-1}} \quad \Delta_{2^m} \quad DH_2 \otimes I_{2^{m-1}}$$

Since $N$ is a power of two, after that (16) can be written in compact form as (17):

$$H_N = \prod_{i=0}^{m-1} I_{2^{m-i-1}} \otimes WH_2 \otimes I_{2^i} \quad I_{2^{m-i-1}} \otimes \frac{1}{2}\Delta_{2^{i+1}} \quad I_{2^{m-i-1}} \otimes DH_2 \otimes I_{2^i} \tag{17}$$

## 3. APPLICATIONS OF THE DEVELOPED ALGORITHM

As an example, and without losing the generality, let us assume the transform length N=16; the $H_N$ matrix can be represented as (18).

$$H_{16} = \prod_{i=0}^{3} I_{2^{3-i}} \otimes WH_2 \otimes I_{2^i} \quad I_{2^{3-i}} \otimes \frac{1}{2}\Delta_{2^{i+1}} \quad I_{2^{3-i}} \otimes DH_2 \otimes I_{2^i} \tag{18}$$

To fully clarify the proposed algorithm, we apply (18) separately for each stage as follows:
For stage one ($i$=0)

$$\begin{aligned} H_{16} &= I_8 \otimes WH_2 \quad I_8 \otimes \frac{1}{2}\Delta_2 \quad I_8 \otimes DH_2 \\ &= I_8 \otimes \frac{1}{2} WH_2\; \Delta_2\; DH_2 \end{aligned} \tag{19}$$

Since $\Delta_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, and $WH_2 = DH_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$. Hence $H_{16} = I_8 \otimes I_2 = I_{16}$. For stage two ($i$=1)

$$\begin{aligned} H_{16} &= I_4 \otimes WH_2 \otimes I_2 \quad I_4 \otimes \frac{1}{2}\Delta_4 \quad I_4 \otimes DH_2 \otimes I_2 \\ &= \left( I_4 \otimes \begin{bmatrix} I_2 & I_2 \\ I_2 & -I_2 \end{bmatrix} \right) \left( I_4 \otimes \frac{1}{2} \begin{bmatrix} I_2 & 0 \\ 0 & I_2 \end{bmatrix} \right) \left( I_4 \otimes \begin{bmatrix} I_2 & I_2 \\ I_2 & -I_2 \end{bmatrix} \right) \\ &= I_{16} \end{aligned} \tag{20}$$

for stages three ($i$=2)

$$H_{16} = I_2 \otimes WH_2 \otimes I_4 \left( I_2 \otimes \frac{1}{2}\Delta_8 \right) I_2 \otimes DH_2 \otimes I_4$$
$$= I_2 \otimes \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.85355 & 0 & 0.35355 & 0 & 0.14645 & 0 & -0.35355 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.35355 & 0 & 0.14645 & 0 & -0.35355 & 0 & 0.85355 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0.14645 & 0 & -0.35355 & 0 & 0.85355 & 0 & 0.35355 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & -0.35355 & 0 & 0.85355 & 0 & 0.35355 & 0 & 0.14645 \end{bmatrix} \tag{21}$$

Following the same procedure for stage four ($i$=3).

$$H_{16} = WH_2 \otimes I_8 \quad \frac{1}{2}\Delta_{16} \quad DH_2 \otimes I_8$$
$$H_{16} =$$

Therefore, it is feasible to calculate transformations using the matrices provided in (18) using the butterfly structure, as illustrated in Figure 1.

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.96194 & 0 & 0 & 0 & 0 & 0 & 0.19134 & 0 & 0.03806 & 0 & 0 & 0 & 0 & 0 & -0.19134 \\
0 & 0 & 0.85355 & 0 & 0 & 0 & 0.35355 & 0 & 0 & 0 & 0.14645 & 0 & 0 & 0 & -0.35355 & 0 \\
0 & 0 & 0 & 0.69134 & 0 & 0.46194 & 0 & 0 & 0 & 0 & 0 & 0.30866 & 0 & -0.46194 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.46194 & 0 & 0.30866 & 0 & 0 & 0 & 0 & 0 & -0.46194 & 0 & 0.69134 & 0 & 0 \\
0 & 0 & 0.35355 & 0 & 0 & 0 & 0.14645 & 0 & 0 & 0 & -0.35355 & 0 & 0 & 0 & 0.85355 & 0 \\
0 & 0.19134 & 0 & 0 & 0 & 0 & 0 & 0.03806 & 0 & -0.19134 & 0 & 0 & 0 & 0 & 0 & 0.96194 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.03806 & 0 & 0 & 0 & 0 & 0 & -0.19134 & 0 & 0.96194 & 0 & 0 & 0 & 0 & 0 & 0.19134 \\
0 & 0 & 0.14645 & 0 & 0 & 0 & -0.35355 & 0 & 0 & 0 & 0.85355 & 0 & 0 & 0 & 0.35355 & 0 \\
0 & 0 & 0 & 0.30866 & 0 & -0.46194 & 0 & 0 & 0 & 0 & 0 & 0.69134 & 0 & 0.46194 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & -0.46194 & 0 & 0.69134 & 0 & 0 & 0 & 0 & 0 & 0.46194 & 0 & 0.30866 & 0 & 0 \\
0 & 0 & -0.35355 & 0 & 0 & 0 & 0.85355 & 0 & 0 & 0 & 0.35355 & 0 & 0 & 0 & 0.14645 & 0 \\
0 & -0.19134 & 0 & 0 & 0 & 0 & 0 & 0.96194 & 0 & 0.19134 & 0 & 0 & 0 & 0 & 0 & 0.03806
\end{bmatrix} \tag{22}$$

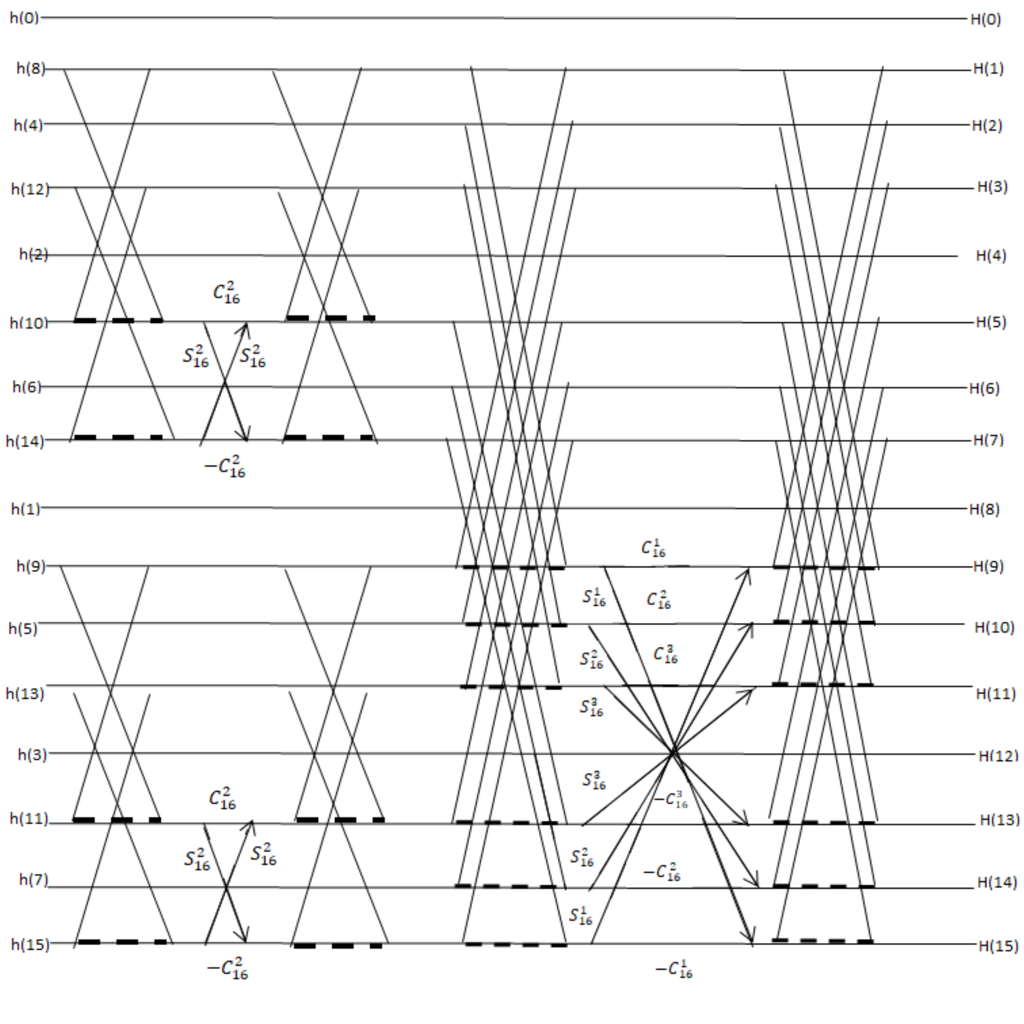



Figure 1. Radix-2 FWHT signal flow diagram when N=16, with (20) multiplications and (50) additions where the solid and dotted lines denote additions and subtractions, respectively

## 4. COMPUTATIONAL COMPLEXITY

According to Figure 1, the proposed algorithm reduces the total number of stages to $(\log_2 N - 2)$ as demonstrated in (19) and (20). Additionally, (21) and (22) show removing the butterflies at the points where $\cos_N^k + \sin_N^k = 1$ , at k=0, $\frac{N}{4}$ . Therefore, we can construct an in-place butterfly for the developed algorithm, as shown in Figure 2.

Hence, the whole transformation satisfies the following:

$$M_N = N(\log_2 N-2)-(N-4) \quad (23)$$

$$A_N = \frac{5}{2}\ N(\log_2 N-2)-(N-4) \quad (24)$$

$A_N$ and $M_N$ denote the overall number of real additions and multiplications. The comparison in the total number of operations number between the FWHT and radix-2 WHT followed by the radix-2 FHT is shown in Table 1. The comparison shows that the FWHT algorithm requires (3N-4) real multiplications and (7.5N-10) real additions less than the existing algorithm. Furthermore, the proposed algorithm was also applied on MATLAB (R2021b), loaded on a laptop computer processor (Intel Core i7), and Windows-10 system to validate and confirm the results of the mathematical operations, as shown in Figures 3(a) and 3(b).

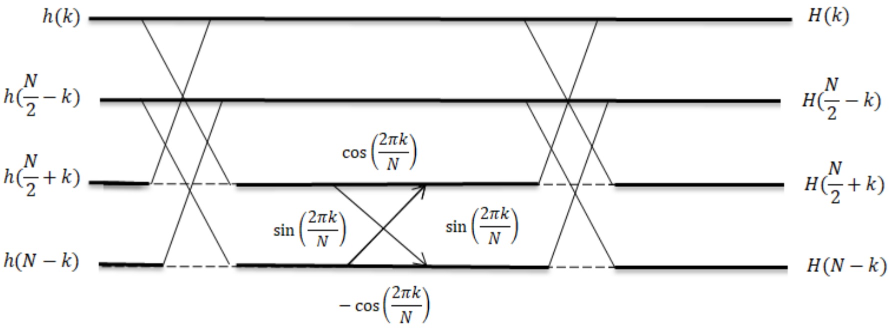
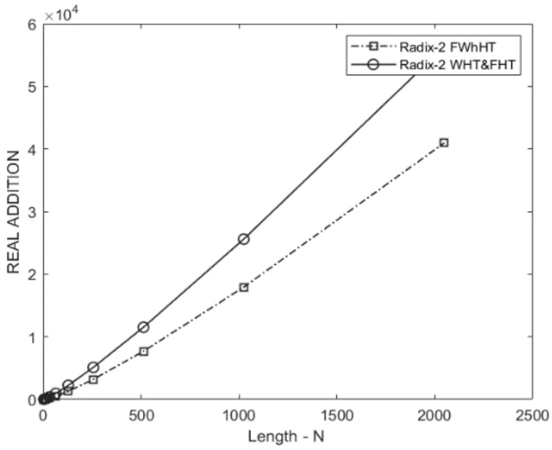


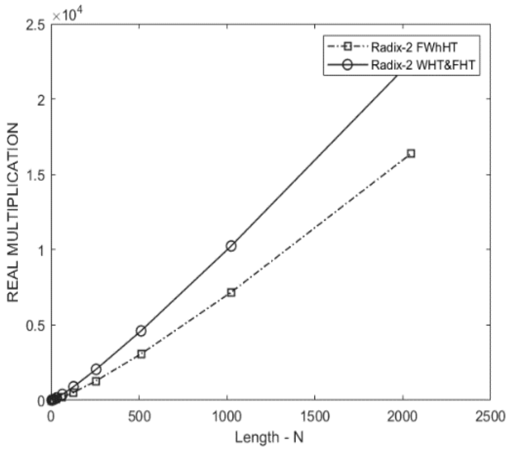

Figure 2. An in-place butterfly of radix-2 FWHT algorithm

Table 1. Comparison of real arithmetic operations

| Transform Length(N) | Radix-2 WHT&FHT | | Radix-2 FWHT | |
|---|---|---|---|---|
| | Multiplications | Additions | Multiplications | Additions |
| 4 | 8 | 20 | 0 | 0 |
| 8 | 24 | 60 | 4 | 10 |
| 16 | 64 | 160 | 20 | 50 |
| 32 | 160 | 400 | 68 | 170 |
| 64 | 384 | 960 | 196 | 490 |
| 128 | 896 | 2240 | 516 | 1290 |
| 256 | 2048 | 5120 | 1284 | 3210 |
| 512 | 4608 | 11520 | 3076 | 7690 |
| 1024 | 10240 | 25600 | 7172 | 17930 |
| 2048 | 22528 | 56320 | 16388 | 40970 |




(a)




(b)

Figure 3. Shows the overall number of real operations (additions and multiplications) for the proposed radix-2 FWHT and WHT+FHT (a) real additions and (b) real multiplications

## 5. CONCLUSION

The paper has been presented an efficient FWHT algorithm as a combination of the fast version of the WHT and the DHT. The developed algorithm is based on sparse matrices factorization using Kronecker product technique. The in-place butterfly structure has been used to implement the newly developed radix-2 FWHT algorithm, and the arithmetic complexity of the proposed algorithm has been computed and investigated in detail. The number of arithmetic operations has been compared with the conventional WHT-FHT method. The result of this comparisons reveals that the proposed algorithm significantly reduced the number of arithmetic operations (multiplications and additions) performed in addition to the simplicity of implementation. The unique characteristics of the transform developed in this paper imply a variety of exciting applications. Although this topic is beyond the scope of this paper, it will be discussed in more depth in a forthcoming publication.

## REFERENCES


[1] A. Dziech, "New orthogonal transforms for signal and image processing," *Applied Sciences*, vol. 11, no. 16, Aug. 2021, doi: 10.3390/app11167433.

[2] H.-S. Ye, N.-R. Zhou, and L.-H. Gong, "Multi-image compression-encryption scheme based on quaternion discrete fractional Hartley transform and improved pixel adaptive diffusion," *Signal Processing*, vol. 175, Oct. 2020, doi: 10.1016/j.sigpro.2020.107652.

[3] P. Zheng and J. Huang, "Efficient encrypted images filtering and transform coding with Walsh-Hadamard transform and parallelization," *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2541–2556, 2018, doi: 10.1109/TIP.2018.2802199.

[4] X. Zhang and Q. Su, "A spatial domain-based color image blind watermarking scheme integrating multilevel discrete Hartley transform," *International Journal of Intelligent Systems*, vol. 36, no. 8, pp. 4321–4345, Aug. 2021, doi: 10.1002/int.22461.

[5] H. A. Leftah and S. Boussakta, "Efficient coded DCT-OFDM system utilizing Walsh-Hadamard transform," in *Wireless Telecommunications Symposium 2012*, Apr. 2012, pp. 1–5, doi: 10.1109/WTS.2012.6266130.

[6] M. Al-Gharabally and A. F. Almutairi, "Frequency-domain subcarrier diversity receiver for discrete Hartley transform OFDM systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, Dec. 2019, doi: 10.1186/s13638-019-1398-0.

[7] E. N. Ayvaz, M. Maraş, M. Gömeç, A. Savaşcıhabeş, and A. Özen, "A novel concatenated LWT and WHT based UFMC waveform design for the next generation wireless communication systems," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 16, no. 5, pp. 743–753, May 2021, doi: 10.1002/tee.23354.

[8] M. Li, H. Zhang, Y. Jin, Z. Wang, and G. Guo, "Parallelizing hartley transform with hadoop for fast detection of glass defects," *Concurrency and Computation: Practice and Experience*, vol. 30, no. 23, pp. 1–11, 2018, doi: 10.1002/cpe.4499.

[9] M. Mohamed Asan Basiri and N. M. Sk, "Discrete orthogonal multi-transform on chip (DOMoC)," *Journal of Signal Processing Systems*, vol. 91, no. 5, pp. 437–457, 2019, doi: 10.1007/s11265-017-1322-y.

[10] M. Kazaryan, M. Shahramanyan, and A. Richter, "Space monitoring of the earth on the presence of solid domestic wastes using a discrete orthogonal transforms," *Serbian Journal of Electrical Engineering*, vol. 14, no. 3, pp. 343–364, 2017, doi: 10.2298/SJEE1703343K.

[11] K. A. Wahid, M. A. Islam, S. S. Shimu, M. H. Lee, and S.-B. Ko, "Hybrid architecture and VLSI implementation of the cosine–fourier–haar transforms," *Circuits, Systems and Signal Processing*, vol. 29, no. 6, pp. 1193–1205, Dec. 2010, doi: 10.1007/s00034-010-9200-x.

[12] B. Said and H. Monir, "Fast walsh–hadamard–fourier transform algorithm," *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5627–5631, 2011.

[13] J. Liu, Q. Xing, X. Yin, X. Mao, and F. Yu, "Pipelined architecture for a Radix-2 fast Walsh–Hadamard–fourier transform algorithm," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 62, no. 11, pp. 1083–1087, Nov. 2015, doi: 10.1109/TCSII.2015.2456371.

[14] N. Jarray, M. Elhajji, and A. Zitouni, "Efficient hybrid DWT-DCT architecture for wireless capsule endoscopy," in *15th International Multi-Conference on Systems, Signals & Devices (SSD)*, Mar. 2018, no. 1, pp. 263–268, doi: 10.1109/SSD.2018.8570369.

[15] A. L. Narayana, B. Prasad, P. R. Kapula, D. Prasad, A. K. Panigrahy, and D. N. V. S. L. S. Indira, "Enhancement in performance of DHTprecoding over WHT for EC companded OFDM in wireless networks," *Applied Nanoscience*, Sep. 2021, doi: 10.1007/s13204-021-02016-x.

[16] D. Gautam and A. Singh, "Resourceful fast discrete Hartley transform to replace discrete Fourier transform with implementation of DHT algorithm for VLSI architecture," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 10, pp. 5290–5298, 2021, doi: 10.17762/turcomat.v12i10.5329.

[17] J. M. Varela, G. Rodriguez, and C. G. Soares, "Comparison study between the Fourier and the Hartley transforms for the real-time simulation of the sea surface elevation," *Applied Ocean Research*, vol. 74, pp. 227–236, May 2018, doi: 10.1016/j.apor.2018.03.002.

[18] M. S. Moreolo, R. Munoz, and G. Junyent, "Novel power efficient optical OFDM based on Hartley transform for intensity-modulated direct-detection systems," *Journal of Lightwave Technology*, vol. 28, no. 5, pp. 798–805, Mar. 2010, doi: 10.1109/JLT.2010.2040580.

[19] G. E. J. Bold, "A comparison of the time involved in computing fast Hartley and fast Fourier transforms," *Proceedings of the IEEE*, vol. 73, no. 12, pp. 1863–1864, 1985, doi: 10.1109/PROC.1985.13381.

[20] M. A. P. Chamikara, A. A. C. A. Jayathilake, and A. A. I. Perera, "Discrete Walsh-Hadamard transform in signal processing related papers discrete Walsh-Hadamard transform in signal processing," *International Journal of Research in Information Technology*, vol. 1, no. 1, pp. 80–89, 2013.

[21] J. Hua, F. Liu, Z. Xu, F. Li, and D. Wang, "A fast realization of new Mersenne number transformation and its applications," *International Journal of Circuit Theory and Applications*, vol. 47, no. 5, pp. 738–752, May 2019, doi: 10.1002/cta.2614.

[22] M. Khurana and H. Singh, "Spiral-phase masked optical image health care encryption system for medical images based on fast Walsh-Hadamard transform for security enhancement," *International Journal of Healthcare Information Systems and Informatics*, vol. 13, no. 4, pp. 98–117, 2018, doi: 10.4018/ijhisi.2018100107.

[23] B. Avci *et al.*, "A novel ACO-OFDM system based on fast Walsh Hadamard transform," in *International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, Jun. 2020, pp. 1–4, doi: 10.1109/ICECCE49384.2020.9179203.
[24] J.-L. Wu, "Block diagonal structure in discrete transforms," *IEE Proceedings E Computers and Digital Techniques*, vol. 136, no. 4, 1989, doi: 10.1049/ip-e.1989.0033.
[25] J. Granata, M. Conner, and R. Tolimieri, "Recursive fast algorithm and the role of the tensor product," *IEEE Transactions on Signal Processing*, vol. 40, no. 12, pp. 2921–2930, 1992, doi: 10.1109/78.175736.
[26] J. R. Johnson, R. W. Johnson, D. Rodriguez, and R. Tolimieri, "A methodology for designing, modifying, and implementing Fourier transform algorithms on various architectures," *Circuits Systems and Signal Processing*, vol. 9, no. 4, pp. 449–500, Dec. 1990, doi: 10.1007/BF01189337.

## BIOGRAPHIES OF AUTHORS

**Suha Suliman Mardan** received the B.Sc. degree in electrical engineering from University of Tikrit, Tikrit, Iraq in 2005. Currently, she is working as a M.Sc student under the supervision of Assistant Prof. Mounir T. Hamood at Tikrit University. Her research interest in the areas of digital signal processing (DSP) and communication systems. She can contact her at email: suha.s.mardan43852@st.tu.edu.iq

**Mounir Taha Hamood** received the B.Sc. degree in electrical engineering from University of Technology, Baghdad, Iraq, in 1990 and the M.Sc. degree in Electronic and Communications Engineering from Al-Nahrain University, Baghdad, Iraq, in 1995. He graduated from Newcastle University, Newcastle upon Tyne, UK in 2012 with the Ph.D. degree in communications and signal processing. His doctoral research was in the development of efficient algorithms for fast computation of discrete transforms. From 2012 to 2016, He was a lecturer in signal processing for communication at Tikrit University. He is currently working as a head of Electrical department and associate professor of signal processing at the Department of Electrical Engineering, College of Engineering, Tikrit University, Tikrit, Iraq. His research interest includes discrete transforms, fast algorithms for digital signal processing in one and multidimensional applications, and communication systems. He can contact him at email: m.t.hamood@tu.edu.iq.