Article

Not peer-reviewed version

Fast Computation for Square Matrix Factorization

Artyom M. Grigoryan *

Posted Date: 17 December 2025

doi: 10.20944/preprints202512.1575.v1

Keywords: QR-factorization; discrete signal-induced heap transformation; Householder reflection

 doi:10.20944/preprints202512.1575.v1

Article

Fast Computation for Square Matrix Factorization

Artyom M. Grigoryan

Department of Electrical and Computer Engineering, The University of Texas at San Antonio, San Antonio, TX 78249, USA; artyom.grigoryan@utsa.edu

Abstract

In this work, we discuss the method of the QR-factorization which is based of the transformations which is called the discrete signal induced heap transformations (DsiHT). These transformations are generated by given signals and can be composed by elementary rotations. The order of processing data, or the path of the transformation, is an important characteristic of it, and the correct choice of such paths can lead to a significant reduction in the operation when calculating the factorization for large matrices. Such paths are called fast paths of the DsiHT, and they define sparse matrices with more zero coefficients than when calculating QR-factorization in the traditional path, that is, when processing data in the natural order $x_0, x_1, x_2, \dots$. For example, in the first stage of the factorization of a 512×512 matrix, a matrix is used with 257,024 zero coefficients of total 262,144 coefficients, when using the fast paths. For comparison, the calculations in the natural order require a 512×512 matrix with only 130,305 zero coefficients it this stage. The effectiveness of the proposed method is illustrated in comparison with the QR-factorization based on the sequence of Householder reflections (or transformations). Examples with the 4×4, 5×5, and 8×8 matrices are described in detail. QR-factorization of complex matrices by complex DsiHTs with the fast paths is also described. The example of the QR-factorization of 256×256 complex matrix is also described and compared with the method of Householder reflections which is used in programming language MATLAB.

Keywords: QR-factorization; discrete signal-induced heap transformation; Householder reflection

1. Introduction

The factorization of a matrix A by an orthogonal (or unitary) matrix Q and a triangular matrix R, which is known as the QR-factorization, or the QR-decomposition, is a fundamental technique which is widely used in various applications in computing and data analysis. This factorization is computationally effective in solving a large linear system $Ax = y$ (as $x = A^{-1}y$), to solve least squares problems [1]-[6], in signal and image processing [7,8], in image watermarking [9,10], in improving air traffic control radar [11], to analyze the stability of linear systems [12,13], in solving linear regression problems in machine learning [14,15], for linear programming algorithms [16], in quantum computing to build effective methods of calculation muti-qubit operations [17]-[20]. The matrix A is presented as the product of two matrices, $A = QR$. The inverse matrix A^{-1} is calculated as $A^{-1} = R^{-1}Q^{-1} = R^{-1}Q^*$. Here, Q^* is the conjugate transpose to Q, and the inverse matrix R^{-1} can be easily calculated, by using an iterative process called back substitution. The QR factorization is a well-known method in linear algebra, but how to implement this in the most efficient way for big data in computing is still a question.

Many methods of QR-factorization have been introduced for real matrices A, and then were modified for complex matrices. We mention the Gram-Schmidt process [21]-[23], the method of Householder reflections (or Householder transformations) [24]-[27], and the Givens rotations [28]-[30]. We consider the QR-factorization with an upper triangular matrix R and will dwell on the factorization which is calculated by the discrete signal-induced heap transformations (DsiHT) [31]. For a real matrix, the DsiHT can be composed by the elementary rotations. Together with these 2×2 basic operations, the DsiHT is determined by the path, or order, of processing the signal components.

 doi:10.20944/preprints202512.1575.v1

The description of these transformations for two particular cases which are called the DsiHT with the weak and strong 2-wheel carriages has been given in [32]. These DsiHT are called the weak and strong DsiHT, respectively. The first transformation operates on the signal $\boldsymbol{x} = (x_0, x_1, x_2, \dots, x_{N-1})$ in the usual way, that is, the 2×2 rotations operate in the order x_0 and x_1, then the renewed x_0 and x_2, and so on. The strong DsiHT processes the signal components in the order x_{N-2} and x_{N-1}, then x_{N-3} with the renewed x_{N-2}, and continuing in this manner, all the signal-generator energy is transferred to the first component of the transform. Every second output of the rotations is zeroed.

The present work is a continuation of the research in the QR-factorization of square matrices. From very beginning much attention was given to the path of the N-point DsiHT, especially for large N. Many paths can be chosen for the transformation and among them we are looking for paths which lead to effective calculation of the DsiHT-based QR-factorization. Each path of the transformation determines the structure of its matrix and the number of zero coefficients in it. Therefore, we are looking for paths that determines sparse matrices of the transformation. Such paths exist and they can be used instead of the paths in the above mentioned weak and strong DsiHTs. We present two such paths, which are call fast paths #3 and #4. The paths for the weak and strong DsiHT are referred to as paths #1 and #2, respectively. The DsiHTs with fast paths are defined by unitary matrices that are sparser than the matrices of the original two DsiHTs. For instance, the 256-point DsiHTs by the fast paths have 63,232 zero coefficients out of 65,536, whereas when using path #1 or #2, the number of zero coefficients is 32,325. It means that fewer operations are required to perform the transformation $H: \boldsymbol{x} \to (\|\boldsymbol{x}\|, 0, \dots, 0)$ and therefore, fewer operations to calculate the QR-factorization.

The main contributions of this work are the following:

- Description of the fast paths for the N-point DsiHT with a sparse matrix. For instance, when $N = 2^r,\ r > 2$, the matrix of the transformation has $N^2 - N(\log_2 N + 1)$ zero coefficients.
- The concept of the DsiHT with new 2×2 complex matrices for the 'complex Givens rotations.'
- The reduced number of multiplications in the QR-factorization.
- Algorithms with fast paths that allow us to perform all real and complex rotation in the QR-factorization only on the adjacent bit planes.
- Encoded tables with $N(N-1)/2$ angles for the real unitary matrix Q. Such tables can be used to generate random unitary $N \times N$ matrices.
- Illustrative examples that show advantage of using the DsiHT with fast paths when comparing with the Householder reflection-based QR-factorization.

The rest of the paper is organized in the following way. In Section II, the concept of the N-point DsiHT is described with examples of different paths for the $N = 4$ case. The comparison with the method of the Householder reflections is also given. Section 3 describes all N-points DsiHTs with fast paths for $N \le 8$. The matrices of the 8-point DsiHTs with five different paths are given and compared with the 8-point Householder reflection. The algorithm for the N-points DsiHT with fast path #4 is also presented in the general case, when $N > 2$. In Section 4, the DsiHT-based QR-factorization is described with the example for a 5×5 real matrix. The concept of the complex DsiHT with the fast paths and different 2×2 basic operations is presented in Section 5. Examples with 5×5 and 256×256 complex matrices are described and compared with the Householder reflection-based QR-factorization calculated in MATLAB.

2. QR-Factorization

In this section, we consider the N-point DsiHT for the case when N is equal to or differs from a power of 2. The transformation requires $(N-1)$ elementary rotations [31]. There are many ways to calculate the transform. As example, we consider the 4-point DsiHTs which are defined by the diagrams shown in Figure 1. The signal is considered real. These diagrams show different orders, or paths, to process the components of the input. In part (a), the components x_0, x_1, x_2, and x_3 of the input signal (generator) $\boldsymbol{x}$ are processed in the natural order. This is the traditional path of

 doi:10.20944/preprints202512.1575.v1

processing the signal. The DsiHT with such a path is called the weak carriage-wheel DsiHT (see [32] for more detail). The first value of the transform is renewed three times. This is why it is denoted by $x_0^{(3)}$. Each 2-point unitary transform T_k, $k = 1,2,3$, is the Givens rotation, or elementary rotation. In the figure, these rotations are depicted as butterflies. In the matrix form, the rotation $T: (x, y) \to \left(\pm\sqrt{x^2 + y^2}, 0\right)$ is described as

$$T\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\vartheta & -\sin\vartheta \\ \sin\vartheta & \cos\vartheta \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \pm\sqrt{x^2 + y^2} \\ 0 \end{bmatrix}. \quad (1)$$

The rotation angle is defined by the inputs as $\vartheta = -\arctan(y/x)$. If $x = 0$, then $\vartheta = \pm\pi/2$. The transform of the generator $\boldsymbol{x} = (x_0, x_1, x_2, x_3)$ is equal to

$$T\boldsymbol{x} = (\pm\|x\|, 0,0,0) = \left(\pm\sqrt{x_0^2 + x_1^2 + x_2^2 + x_3^2}, 0,0,0\right). \quad (2)$$

Thus, the energy $E[\boldsymbol{x}] = \|\boldsymbol{x}\|$ of the generator is transformed to the first component. The angles ϑ_k, $k = 0,1,2$, are calculated from this generator. The set $A_x = \{\vartheta_1, \vartheta_2, \vartheta_3\}$ is called the angular representation of the signal-generator. Once the angles have been calculated, the transform of an input $\boldsymbol{z} = (z_0, z_1, z_2, z_3)$ is calculated by using the same path,

$$T_1: (z_0, z_1) \to \left(z_0^{(1)}, z_1'\right),\ T_2: \left(z_0^{(1)}, z_2\right) \to \left(z_0^{(2)}, z_2'\right),\ T_3: \left(z_0^{(2)}, z_3\right) \to \left(z_0^{(3)}, z_3'\right). \quad (3)$$

The result of the transformation is the signal $T\boldsymbol{z} = \left(z_0^{(3)}, z_1', z_2', z_3'\right)$. The 4-point DsiHT is a unitary transformation. Therefore, the energy of the signal is preserved, $E[\boldsymbol{z}] = \|\boldsymbol{z}\| = \|T(\boldsymbol{z})\|$.

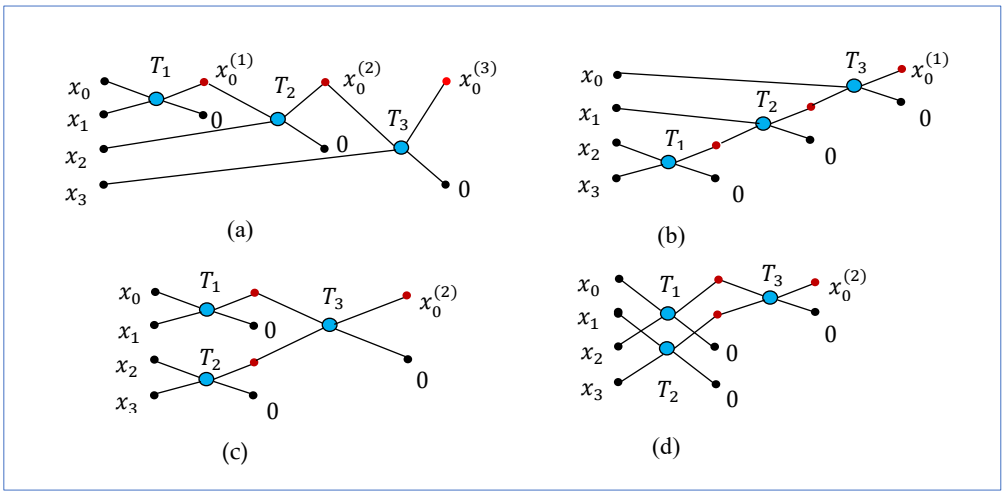


Figure 1. Four diagrams for calculating the 4-point DsiHT.

Path #2 which is shown in the diagram in Figure 1 part (b) corresponds to the DsiHT which is called the strong carriage-wheel DsiHT [32]. Here, processing data starts with the last component x_3 of the signal. The first component x_0 is updated once at the end of the computation. Now, we consider two paths of the 4-point DsiHT shown in parts (c) and (d) in Figure 1. There are two different partitions of data on the first stage of computation. Two rotations T_1 and T_2 can be calculated in parallel. Total number of rotations, or butterflies, is equal to 3. We will call paths #3 and #4 of calculation shown in part (c) and (d), respectively, the fast paths. Of greatest interest are these two paths, which allow the calculation of the transform to be performed in two stages (or the minimum number of stages). Here, we also mention the following. Methods of QR-factorization by rotations are widely used in quantum computation to build quantum circuits for multi-qubit operations [33,34]. All rotations are required to operate on adjacent bit-planes (BP). Such bit-planes differ only in one bit. In the above case, when the signals are 4-point vectors, we can consider that the components $x_0, x_1, x_2,$ and x_3 lie on the bit-planes 0,1,2, and 3, that is 00, 01, 10, and 11, respectively. Three butterflies in the diagrams with fast paths operate only on adjacent BPs. For instance, the second butterfly T_2 operates on BPs (10) and (11) when using path #3, and on BP (01) and (11), when

 doi:10.20944/preprints202512.1575.v1

using path #4. For comparison, the butterfly T_3 in the diagram in part (a) operates on BPs (00) and (11), which are not adjacent BPs. Also, one can see that in the diagram in part (b), the second rotation T_2 operates on non-adjacent BPs (01) and (10). In general, for any integer $N > 2$, we call the path of the N-point DsiHT the fast path, if all butterflies (rotations) operate only on adjacent BPs.

Example 1. *Let the generator be* $\boldsymbol{x} = (1,3,-2,5)$ *with the energy* $E[\boldsymbol{x}] = \sqrt{39}$. *The matrices of the 4-point DsiHTs generated by* $\boldsymbol{x}$ *and four paths in Figure 1 are the following:*

$$H_{4;\#1} = \begin{bmatrix} 0.1601 & 0.4804 & -0.3203 & 0.8006 \\ -0.9487 & 0.3162 & 0 & 0 \\ 0.1690 & 0.5071 & 0.8452 & 0 \\ -0.2140 & -0.6419 & 0.4280 & 0.5991 \end{bmatrix}, \quad \det H_{4;\#1} = 1, \tag{4}$$

$$H_{4;\#2} = \begin{bmatrix} 0.1601 & 0.4804 & -0.3203 & 0.8006 \\ -0.9871 & 0.0779 & -0.0520 & 0.1299 \\ 0 & 0.8736 & 0.1807 & -0.4519 \\ 0 & 0 & 0.9285 & 0.3714 \end{bmatrix}, \quad \det H_{4;\#2} = 1, \tag{5}$$

$$H_{4;\#3} = \begin{bmatrix} 0.1601 & 0.4804 & -0.3203 & 0.8006 \\ -0.9487 & 0.3162 & 0 & 0 \\ 0.2727 & 0.8181 & 0.1881 & -0.4702 \\ 0 & 0 & 0.9285 & 0.3714 \end{bmatrix}, \quad \det H_{4;\#3} = 1, \tag{6}$$

$$H_{4;\#4} = \begin{bmatrix} 0.1601 & 0.4804 & -0.3203 & 0.8006 \\ -0.4176 & 0.1842 & 0.8351 & 0.3070 \\ 0.8944 & 0 & 0.4472 & 0 \\ 0 & -0.8575 & 0 & 0.5145 \end{bmatrix}, \quad \det H_{4;\#4} = 1. \tag{7}$$

The transforms $H_{4;\#k}\boldsymbol{x}' = \left(\sqrt{39},0,0,0\right)'$, for $k = 1{:}4$. Each path determines the structure of the matrix. This can be well seen by the location and the number of zero coefficients in the above matrices. The number of zero coefficients in the first two matrices is equal to 3, and in the last two matrices this number is 4. The first rows in these matrices are the generator $\boldsymbol{x}$ normalized to the coefficient of 0.1601, that is, $(1,3,-2,5) \times 0.1601$. All numbers are given with an accuracy of 4 decimal places. The angles $\vartheta_k, k = 1,2,3,$ (in degrees) for these transformations are given in Table 1.

Table 1. The angles of rotation in the matrices of the 4-point DsiHT.

	$\boldsymbol{\vartheta_1}$	$\boldsymbol{\vartheta_2}$	$\boldsymbol{\vartheta_3}$
$H_{4;\#1}$	$-71.5651°$	$32.3115°$	$-53.1913°$
$H_{4;\#2}$	$68.1986°$	$60.8784°$	$-80.7857°$
$H_{4;\#3}$	$-71.5651°$	$68.1986°$	$59.5777°$
$H_{4;\#4}$	$63.4349°$	$-59.0362°$	$-69.0191°$

For comparison, we consider the Householder transformation matrix composed by the signal $\boldsymbol{x}$, as follows [4,25]:

$$P_4 = I_4 - 2\frac{vv'}{\|v\|} = \begin{bmatrix} -0.1601 & -0.4804 & 0.3203 & -0.8006 \\ -0.4804 & 0.8011 & 0.1326 & -0.3315 \\ 0.3203 & 0.1326 & 0.9116 & 0.2210 \\ -0.8006 & -0.3315 & 0.2210 & 0.4475 \end{bmatrix}, \quad \det P_4 = -1, \tag{8}$$

where I_4 is the 4×4 identity matrix and the Householder vector $\boldsymbol{v}$ is calculated as $v = \boldsymbol{x} + \|\boldsymbol{x}\|(1,0,0,0) = \left(1 + \sqrt{39},3,-2,5\right)'$. This matrix is symmetric (without zero coefficients) and orthogonal, and $P_4\boldsymbol{x}' = (-\sqrt{39},0,0,0)'$.

All these five matrices are different. As an example, we consider the input signal $\boldsymbol{z} = (1,-3,2,5)'$. Then, we obtain the following five different transforms:

$$H_{4;\#1}\boldsymbol{z} = (2.081666, -1.897367, 0.338062, 5.563486)', \tag{9}$$

$$H_{4;\#2}\boldsymbol{z} = (2.081666, -0.675382, -4.518564, 3.713907)', \tag{10}$$

$$H_{4;\#3}\boldsymbol{z} = (2.081666\ , -1.897367, -4.156148, 3.713907)', \tag{11}$$

 doi:10.20944/preprints202512.1575.v1

$$H_{4;\#4}\boldsymbol{z} = (2.081666, 2.235191, 1.788854, 5.144958)', \quad (12)$$

$$P_4\boldsymbol{z} = (-2.081666, -4.276053, 2.850702, 2.873246)'. \quad (13)$$

3. N-Point DsiHT, when $N \leq 8$

In this section, we describe N-point DsiHTs, starting from the number $N = 8$ and reducing it to 3. For each of these transformations, we consider two fast paths, which are similar to the fast paths in Figure 1 in parts (c) and (d). First, we consider the 8-point DsiHTs with paths given in Figure 2. Seven butterflies are used in each of the diagrams shown in parts (a) and (b) and all butterflies operate on adjacent BPs. Four butterflies are used in the first stage, and they can be calculated in parallel. Two butterflies operate in the second stage, and one butterfly in the last stage. It is difficult to say which of these two paths is better. One can note in the diagram in part (b) that the calculations after Stage 1 represent the 4-point DsiHT described in Figure 1(d).

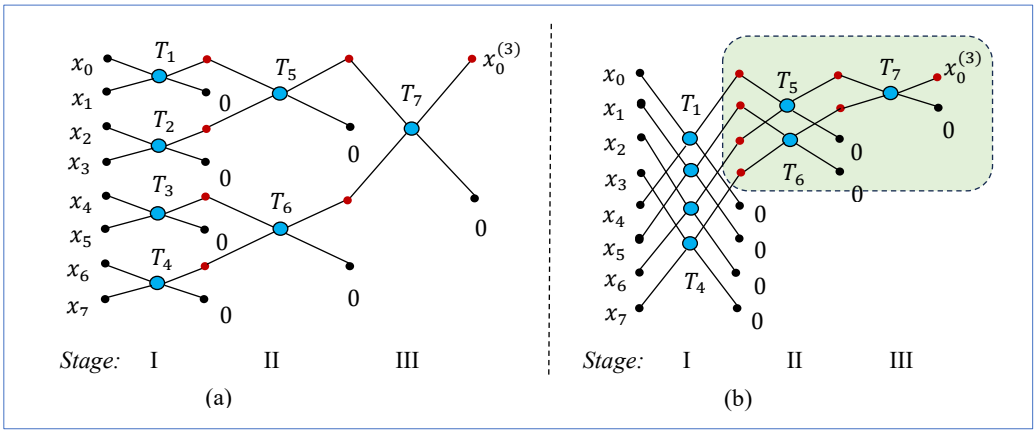


Figure 2. The diagrams for calculating the 8-point DsiHT with (a) path #3 and (b) path #4.

The paths which are similar to paths in Figure 1(a) and (b) can also be used for the 8-point DsiHT. It should be noted that, other fast paths exist for calculating the 8-point DsiHT. As an example, we consider the transformation with the diagram shown in Figure 3. The larger the value of N, the more paths can be found for computing the N-point DsiHT.

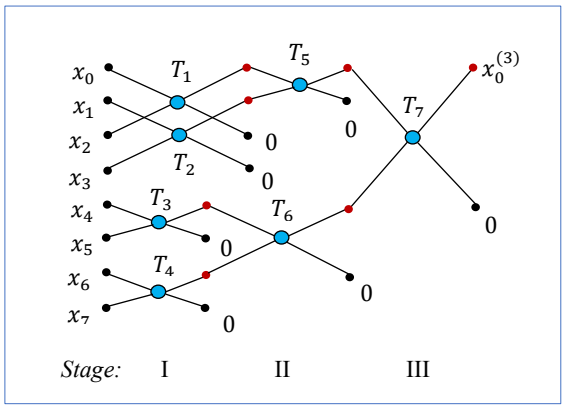


Figure 3. The diagram for calculating the 8-point DsiHT with path #5.

Example 2. *Let the generator be* $\boldsymbol{x} = (1,3,2,4,-2,-1,3,5)'$ *with the energy* $E[\boldsymbol{x}] = \sqrt{69}$. *The matrices of the 8-point DsiHTs generated by* $\boldsymbol{x}$ *and paths #1, 2, 3, and 4 are the following:*

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 December 2025 doi:10.20944/preprints202512.1575.v1

$$H_{8;\#1} = \begin{bmatrix} 0.1204 & 0.3612 & 0.2408 & 0.4815 & -0.2408 & -0.1204 & 0.3612 & 0.6019 \\ -0.9487 & 0.3162 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.1690 & -0.5071 & 0.8452 & 0 & 0 & 0 & 0 & 0 \\ -0.1952 & -0.5855 & -0.3904 & 0.6831 & 0 & 0 & 0 & 0 \\ 0.0626 & 0.1879 & 0.1252 & 0.2505 & 0.9393 & 0 & 0 & 0 \\ 0.0290 & 0.0870 & 0.0580 & 0.1160 & -0.0580 & 0.9856 & 0 & 0 \\ -0.0764 & -0.2293 & -0.1529 & -0.3058 & 0.1529 & 0.0764 & 0.8919 & 0 \\ -0.0907 & -0.2722 & -0.1815 & -0.3630 & 0.1815 & 0.0907 & -0.2722 & 0.7985 \end{bmatrix}, \quad (14)$$

$$H_{8;\#2} = \begin{bmatrix} 0.1204 & 0.3612 & 0.2408 & 0.4815 & -0.2408 & -0.1204 & 0.3612 & 0.6019 \\ -0.9927 & 0.0438 & 0.0292 & 0.0584 & -0.0292 & -0.0146 & 0.0438 & 0.0730 \\ 0 & -0.9315 & 0.0947 & 0.1895 & -0.0947 & -0.0474 & 0.1421 & 0.2368 \\ 0 & 0 & -0.9655 & 0.1404 & -0.0702 & -00351 & 0.1053 & 0.1755 \\ 0 & 0 & 0 & 0.8421 & 0.1727 & 0.0864 & -02591 & -0.4318 \\ 0 & 0 & 0 & 0 & -0.9473 & 0.0541 & -0.1624 & -0.2707 \\ 0 & 0 & 0 & 0 & 0 & 0.9856 & 0.0870 & 0.1449 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.8575 & 0.5145 \end{bmatrix}, \quad (15)$$

$$H_{8;\#3} = \begin{bmatrix} 0.1204 & 0.3612 & 0.2408 & 0.4815 & -0.2408 & -0.1204 & 0.3612 & 0.6019 \\ -0.9487 & 0.3162 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.2582 & -0.7746 & 0.2582 & 0.5164 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.8944 & 0.4472 & 0 & 0 & 0 & 0 \\ 0.1373 & 0.4118 & 0.2745 & 0.5490 & 0.2112 & 0.1056 & -0.3168 & -0.5279 \\ 0 & 0 & 0 & 0 & -0.4472 & 0.8944 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.8351 & 0.4176 & 0.1842 & 0.3070 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.8575 & 0.5145 \end{bmatrix}, \quad (16)$$

$$H_{8;\#4} = \begin{bmatrix} 0.1204 & 0.3612 & 0.2408 & 0.4815 & -0.2408 & -0.1204 & 0.3612 & 0.6019 \\ -0.2026 & 0.2146 & -0.4053 & 0.2861 & 0.4053 & -0.0715 & -0.6079 & 0.3576 \\ -0.3801 & 0 & 0.2924 & 0 & 0.7601 & 0 & 0.4385 & 0 \\ 0 & -0.8506 & 0 & 0.2766 & 0 & 0.2835 & 0 & 0.3458 \\ 0.8944 & 0 & 0 & 0 & 0.4472 & 0 & 0 & 0 \\ 0 & 0.3162 & 0 & 0 & 0 & 0.9487 & 0 & 0 \\ 0 & 0 & -0.8321 & 0 & 0 & 0 & 0.5547 & 0 \\ 0 & 0 & 0 & -0.7809 & 0 & 0 & 0 & 0.6247 \end{bmatrix}. \quad (17)$$

If we consider the 8-point DsiHT with path #5 shown in Figure 3, we obtain the following matrix of the transformation:

$$H_{8;\#5} = \begin{bmatrix} 0.1204 & 0.3612 & 0.2408 & 0.4815 & -0.2408 & -0.1204 & 0.3612 & 0.6019 \\ -0.4082 & 0.2449 & -0.8165 & 0.3266 & 0 & 0 & 0 & 0 \\ -0.8944 & 0 & 0.4472 & 0 & 0 & 0 & 0 & 0 \\ 0 & -0.8000 & 0 & 0.6000 & 0 & 0 & 0 & 0 \\ -0.1373 & -0.4118 & -0.2745 & -0.5490 & -0.2112 & -0.1056 & 0.3168 & 0.5279 \\ 0 & 0 & 0 & 0 & 0.4472 & -0.8944 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.8351 & 0.4176 & 0.1842 & 0.3070 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.8575 & 0.5145 \end{bmatrix}. \quad (18)$$

Each path defines the structure of the transformation matrix. The first rows in these matrices are the normalized generator $\boldsymbol{x}$, that is, $(1,3,2,4-2,-1,3,5)\times 0.1204$, and $H_{8;\#n}\boldsymbol{x} = \left(\sqrt{69},0,0,\ldots,0\right)', n = 1{:}5$. The angles $\vartheta_k, k = 1{:}7$, of rotations in these transformations are given in Table 2 (in degrees). The number of zero coefficients in these matrices is also shown in the table (in the last column)

Table 2. Angles of rotations in the 8-point DsiHTs by paths #1, 2, 3, 4, and 5.

	ϑ_1	ϑ_2	ϑ_3	ϑ_4	ϑ_5	ϑ_6	ϑ_7	#0
$H_{8;\#1}$	−71.5651°	−32.3115°	−46.9113°	20.0596°	9.7315°	−26.8892°	−37.0082°	21
$H_{8;\#2}$	−59.0362°	80.2685°	−71.3216°	57.3599°	−74.9075°	−68.6660°	−83.0856°	21
$H_{8;\#3}$	−71.5651°	−63.4349°	−26.5651°	−59.0362°	−54.7356°	69.0191°	48.7474°	32
$H_{8;\#4}$	63.4349°	18.4349°	−56.3099°	−51.3402°	−58.1939°	−63.7169°	−59.2859°	32
$H_{8;\#5}$	−63.4349°	−53.1301°	153.4349°	−59.0362°	−65.9052°	−69.0191°	−48.7474°	32

 doi:10.20944/preprints202512.1575.v1

Now, we consider the Householder transformation matrix calculated as $P_8 = I_8 - 2\boldsymbol{v}\boldsymbol{v}'/\|\boldsymbol{v}\|$ and equal to

$$P_8 = \begin{bmatrix} -0.1204 & -0.3612 & -0.2408 & -0.4815 & 0.2408 & 0.1204 & -0.3612 & -0.6019 \\ -0.3612 & 0.8836 & -0.0776 & -0.1552 & 0.0776 & 0.0388 & -0.1164 & -0.1940 \\ -0.2408 & -0.0776 & 0.9483 & -0.1035 & 0.0517 & 0.0259 & -0.0776 & -0.1294 \\ -0.4815 & -0.1552 & -0.1035 & 0.7930 & 0.1035 & 0.0517 & -0.1552 & -0.2587 \\ 0.2408 & 0.0776 & 0.0517 & 0.1035 & 0.9483 & -0.0259 & 0.0776 & 0.1294 \\ 0.1204 & 0.0388 & 0.0259 & 0.0517 & -0.0259 & 0.9871 & 0.0388 & 0.0647 \\ -0.3612 & -0.1164 & -0.0776 & -0.1552 & 0.0776 & 0.0388 & 0.8836 & -0.1940 \\ -0.6019 & -0.1940 & -0.1294 & -0.2587 & 0.1294 & 0.0647 & -0.1940 & 0.6766 \end{bmatrix}. \quad (19)$$

There is no zero coefficient in the matrix. The determinant of this matrix is equals to -1 and the transform $P_8\boldsymbol{x}' = (-\sqrt{69}, 0, \dots, 0,0)'$. The vector v is calculated as $v = x + \|\boldsymbol{x}\|(1,0,\dots,0,0)' = \left(1 + \sqrt{69}, 3,2,4,-2,-1,3,5\right)'$.

One can notice that the number of zero coefficients in the matrices $H_{8;\#3}$ and $H_{8;\#4}$ (and $H_{8;\#5}$) is much greater than in the matrices $H_{8;\#1}$ and $H_{8;\#2}$. These numbers are equal to 32 and 21, respectively. The number, $z_{\#1,2}(N)$, of zero coefficients in the matrices $H_{N;\#1}$ and $H_{N;\#2}$ is the same. The same for the numbers, $z_{\#3,4}(N)$, of zero coefficients in the matrices $H_{N;\#3}$ and $H_{N;\#4}$. In the case when $N = 2^r, r \geq 2$, these numbers are calculated as follows:

$$z_{\#1,2}(N) = \frac{(N-1)(N-2)}{2}, \qquad z_{\#3,4}(N) = N^2 - N(\log_2 N + 1). \quad (20)$$

Figure 4 shows the graphs of these functions in percentage terms, for $N = 2^r$, for $r = 2{:}12$. That is, the number of zeros as a percentage of the matrix size 4^r is calculated for each of these functions,

$$p_{\#1,2}(r) = z_{\#1,2}(2^r)/4^r \times 100\% \text{ and } p_{\#3,4}(r) = z_{\#3,4}(2^r)/4^r \times 100\%. \quad (21)$$

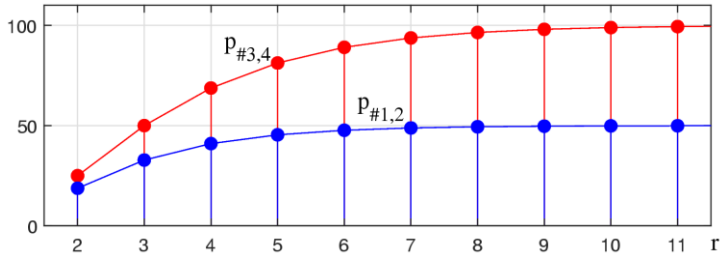


Figure 4. The graphs of the number of zero coefficients in the matrices of the 2^r-point DsiHTs.

For comparison, the first nine values of these numbers are given in Table 3. The table also shows the difference Δz between the numbers $z_{\#3,4}$ and $z_{\#1,2}$ and their percentage to all N^2 coefficients in the matrices.

Table 3. Comparison of four matrices of the N-point DsiHTs, when N is a power of 2.

N	$z_{\#1,2}$	$z_{\#3,4}$	Δz	N^2	$z_{\#1,2}/N^2 \times 100\%$	$z_{\#3,4}/N^2 \times 100\%$
4	3	4	1	16	18.75%	25%
8	21	32	11	64	32.81%	50%
16	105	176	71	256	41.01%	68.75%
32	465	832	367	1,024	45.41%	81.25%
64	1,953	3,648	1,695	4,096	47.68%	89.06%
128	8,001	15,360	7,359	16,384	48.83%	93.75%
256	32,385	63,232	30,847	65,536	49.41%	96.48%
512	**130,305**	**257,024**	126,719	**262,144**	49.70%	98.05%
1024	522,753	1,037,312	514,559	1,048,576	49.85%	98.93%
2048	2,094,081	4,169,728	2,075,647	4,194,304	49.93%	99.41%

For $N = 1024$, the difference of zero coefficients $z_{\#3,4}$ and $z_{\#1,2}$ is equal to 514,559. When multiplying matrices of the 1024-point DsiHT by a 1024-D vector $\boldsymbol{z}$, the calculation requires $1024^2 -$

 doi:10.20944/preprints202512.1575.v1

$z_{\#3,4}(1024) = 11{,}264$ multiplications, if using the transforms with paths #3 and 4. For matrices with paths #1 and #2, the calculation requires $1024^2 - z_{\#1,2}(1024) = 525{,}823$ multiplications, or more than 46 times. In the general case of $N = 2^r, r > 2$, the number of multiplications when multiplying the matrices $H_{\#3,4}$ by the vector $\boldsymbol{z}$ is estimated as

$$\mu_{\#3,4}(N) = N^2 - z_{\#3,4}(N) = N(\log_2 N + 1), \tag{22}$$

and when multiplying $H_{\#1,2}$, the number of multiplications is estimated as

$$\mu_{\#1,2}(N) = N^2 - z_{\#1,2}(N) = \frac{1}{2}(N^2 + 3N - 2). \tag{23}$$

The number of operations saved when using the fast paths #3 and #4 is

$$\Delta z(N) = z_{\#3,4}(N) - z_{\#1,2}(N) = \frac{1}{2}\left(N^2 - N(2r-1)\right) + 1. \tag{24}$$

These numbers are shown in the column number 4 in Table 3.

Next, we show that the diagrams of the 8-point DsiHT, which are shown in Figure 2 for two different paths, can be used and simplified when calculating the 7, 6, and 5-point DsiHTs, by removing the last inputs on the diagrams.

3.1. The **7**-Point DsiHT

Let us consider the $N = 7$ case and the signal-generator $\boldsymbol{x} = (x_0, x_1, \ldots, x_6)$. It is possible to consider the extended zero padded 8-D vector $\tilde{\boldsymbol{x}} = (x_0, x_1, \ldots, x_6, 0)$ and use the diagrams in Figure 2, after removing the rotations number 4 and then changing the numbers of the next three rotations. The corresponding diagrams are shown in Figure 5. On stage 1, three rotations are used. The total number of rotations is equal to six.

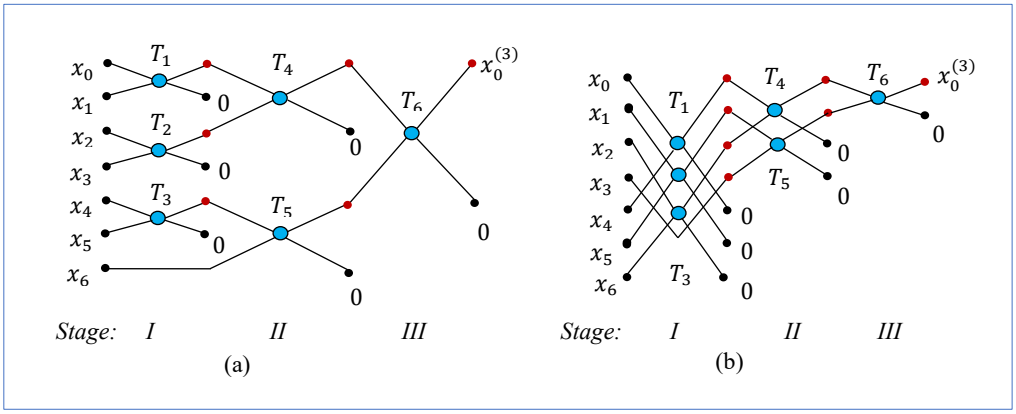


Figure 5. The diagrams for calculating the 7-point DsiHT with (a) path #3 and (b) path #4.

3.2. The **6**-Point DsiHT

The diagrams for calculating the 6-point DsiHTs by paths #3 and #4 are shown in Figure 6 in parts (a) and (b), respectively. For that, the diagrams in Figure 5 were used, after removing the last component x_6 from the input. In these two diagrams, the number of rotations on stages 1 and 2 are different. The total number of rotations is five.

 doi:10.20944/preprints202512.1575.v1

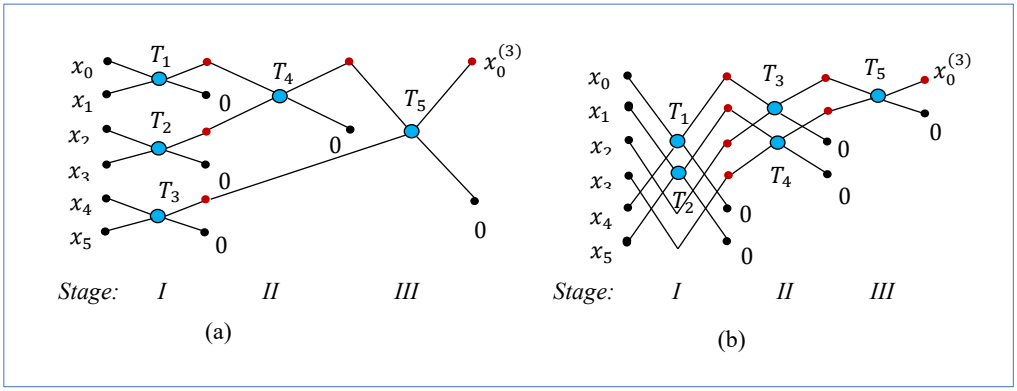


Figure 6. The diagrams for calculating the 6-point DsiHT with (a) path #3 and (b) path #4.

3.3. The **5***-Point DsiHT*

The last two simplifications of the diagrams of the 8-point DsiHT for the 5-point DsiHT are shown in Figure 7. Total four butterflies are used in both diagrams.

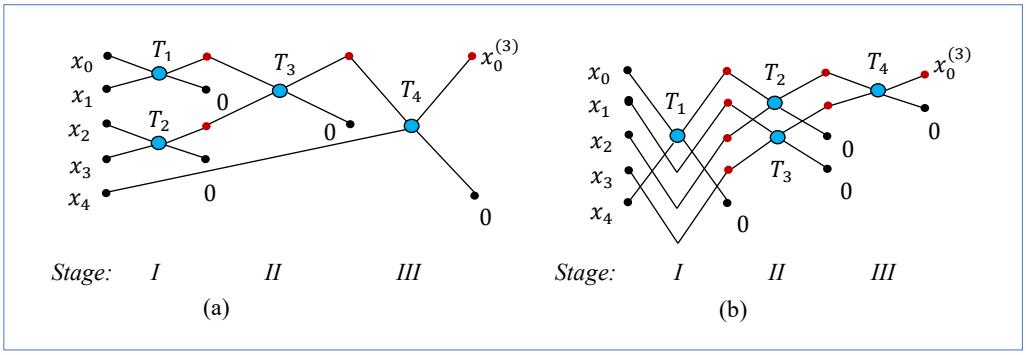


Figure 7. The diagrams for calculating the 5-point DsiHT with (a) path #3 and (b) path #4.

3.4. The **3***-Point DsiHT*

The diagrams of the 4-point DsiHTs are given in Figure 1. For the 3-point DsiHT, there are only two paths can be considered. The diagrams with these paths are shown in Figure 8. Two butterflies are used for these transforms.

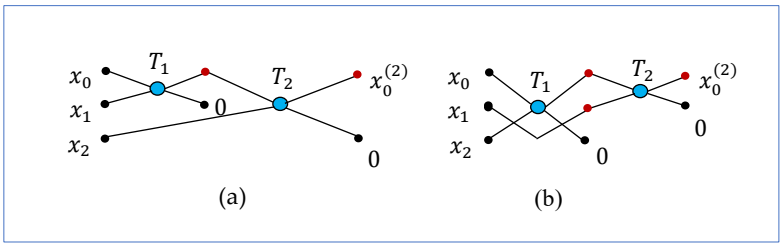


Figure 8. The diagrams for calculating the 3-point DsiHT with (a) path #3 and (b) path #4.

3.5. Algorithm of the DsiHT by Path #4

We stand on path #4, for any order N of the DsiHT. The algorithm of the transformation with this path can be described as follows. Let us consider the smallest power of two, $M = 2^n \geq N$. The calculation of the transform generated by the signal $\boldsymbol{x} = (x_0, x_1, \dots, x_{N-1})$ can be performed in two steps as follows:

1. The Givens rotations are applied to the pairs

$$R_{\vartheta_k}: (x_k, x_{k+M/2}) \to (x'_k, 0), \qquad k = 0,1, \dots, \left(N - \frac{M}{2} - 1\right). \tag{25}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 December 2025 doi:10.20944/preprints202512.1575.v1

We will receive the signal $\boldsymbol{y} = \left(\underbrace{x'_0, x'_1, \dots, x'_{N-M/2-1}}, \underbrace{x_{N-M/2}, \dots, x_{M/2-1}}, \underbrace{0,0, \dots, 0}\right)$.

2. The $M/2$-point DsiHT with the fast path #4 is applied to the $M/2$-point signal

$$\boldsymbol{y_1} = \left(\underbrace{x'_0, x'_1, \dots, x'_{N-M-1}}, \underbrace{x_{N-M/2}, \dots, x_{M/2-1}}\right). \tag{26}$$

The total number of rotations is equal to $(N-1)$.

As an example, we consider the diagram of the 7-point DsiHT, which is shown in Figure 5 part (b), when $N = 7$. The first three rotations are applied to the pairs (x_0, x_4), (x_1, x_5), and (x_2, x_6). Then, the first outputs of these rotations $T_k: (x_k, x_{k+4}) \to (x'_k, 0), k = 0,1,2,$ together with the input component x_3 are used as the input signal, $\boldsymbol{y} = \left(\underbrace{x'_0, x'_1, x'_2}, x_3\right)$, for the 4-point DsiHT. The composition of the 7-point DsiHT is described by the following six matrices of rotation:

$$H_{7;4} = R_{\vartheta_{6;(0,1)}} R_{\vartheta_{5;(1,3)}} R_{\vartheta_{4;(0,2)}} R_{\vartheta_{3;(2,6)}} R_{\vartheta_{2;(1,5)}} R_{\vartheta_{1;(0,4)}}. \tag{27}$$

Here, $R_{\vartheta_{k;(i,j)}}$, when $j \neq i$, denotes the matrix composed by the 7×7 identity matrix $[a_{n,m}; n, m = 0{:}6]$ and coefficients of the matrix rotation R_{ϑ_k} at the points (i,i), (i,j), (j,i), and (j,j). Thus,

$$H_{7;\#4} = \begin{bmatrix} c_6 & -s_6 & 0 & 0 & 0 & 0 & 0 \\ s_6 & c_6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_5 & 0 & -s_5 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & s_5 & 0 & c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_4 & 0 & -s_4 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_4 & 0 & c_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$
$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_3 & 0 & 0 & 0 & -s_3 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & s_3 & 0 & 0 & 0 & c_3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 & 0 & -s_2 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & s_2 & 0 & 0 & 0 & c_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 & 0 & 0 & 0 & -s_1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ s_1 & 0 & 0 & 0 & c_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{28}$$

where $c_k = \cos\vartheta_k$ and $s_k = \sin\vartheta_k$, $k = 1{:}6$. The transform of the signal-generator $\boldsymbol{x}$ is the vector

$$H_7\boldsymbol{x} = \left(x_0^{(k)}, 0,0,0,0,0,0\right)' = \left(\sqrt{x_0^2 + x_1^2 + \cdots + x_6^2}, 0,0,0,0,0,0\right)'. \tag{29}$$

3.6. Number of Zero Coefficients in the DsiHT Matrices

For the N-point DsiHT calculated by paths #3 and 4, the number $z(N)$ of zero coefficients is estimated as follows. If $2^r \leq N < 2^{r+1}$, where $r \geq 2$, the following recurrence formula is valid:

$$z(N+1) = z(N) + 2N - (r+4). \tag{30}$$

In other words, in the $(N+1) \times (N+1)$ matrix of the $(N+1)$-point DsiHT there is $2N - (r+4)$ more zero coefficients than in the matrix of the N-point DsiHT. If we denote $N = 2^r + n$, where $n \in \{1,2, \dots, 2^r\}$, then the number $z(N)$ can be calculated as

$$z(N) = z_{\#3,4}(2^r) + 2^{r+1}n + n(n-r-3) = 4^r + 2^r(2n-r-1) + n(n-r-3). \tag{31}$$

If $N = 2^r$, then $z(N) = z_{\#3,4}(N)$ calculated by Equation (20). A few values of these numbers are shown in Table 4.

Table 4. The number the zeros and operations of multiplication by matrices of the N-point DsiHT.

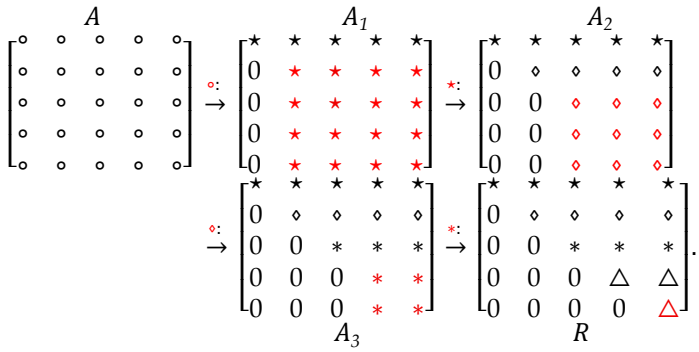
N	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$z(N)$	1	4	8	14	22	32	43	56	71	88	107	128	151	176
$\mu(N)$	8	12	17	22	27	32	38	44	50	56	62	68	74	80
N^2	9	16	25	36	49	64	81	100	121	144	169	196	225	256

 doi:10.20944/preprints202512.1575.v1

The number of multiplications when multiplying the matrices H_N by the N-dimensional vector $\boldsymbol{z}$ can be estimated as $\mu(N) = N^2 - z(N)$. These numbers are also given in Table 4 along with the maximum number of multiplications N^2 for comparison. Note that for the N-point DsiHT with path #1, the number of zero coefficients is calculated by $z_{1,2}(N) = (N^2 - 3N + 2)/2$, for any $N \geq 3$.

4. DsiHT-Based QR Factorization

In this section, we describe the QR-factorization of a square matrix A of size $N \times N, N \geq 3$, by the Givens rotations. The unitary matrix A is considered with real coefficients. In the QR-factorization, $(N-1)$ DsiHTs are used. This factorization is illustrated below for a 5×5 unitary matrix,



(32)

The first 5-point DsiHT, H_5, is generated by the first column of the matrix A and then it is applied to each of its columns. Four zero coefficients will be obtained in the new matrix A_1 in its first column, as shown above. A similar transform will be applied on the 4×4 submatrix (highlighted in red) of A_1. Namely, the 4-point DsiHT, H_4, will be generated by the last four components of the second column of the matrix A_1. This transform will be applied to each column of the 4×4 submatrix. In the new matrix A_2, another three zero coefficients in the 2nd column will be obtained. Then, the 3-point DsiHT, H_3, will be generated by the first column of the 3×3 sub-matrix of A_2 and applied on this sub-matrix. We will obtain a new matrix A_3 with 9 zero coefficients, as shown in Equation (32). The last 2-point DsiHT, H_2, will be generated by the last two coefficients of the 4th column of A_3 and applied to its 2×2 sub-matrix. Thus, the matrix triangularization is completed as

$$R = (I_3 \oplus H_2)(I_2 \oplus H_3)(1 \oplus H_4)H_5 A. \tag{33}$$

Here, I_n is the identity $n \times n$ matrix. If we denote the unitary matrix $Q_1 = (I_3 \oplus H_2)(I_2 \oplus H_3)(1 \oplus H_4)H_5$ and its inverse

$$Q = Q_1^{-1} = Q_1' = H_5'(1 \oplus H_4')(I_2 \oplus H_3')(I_3 \oplus H_2'), \tag{34}$$

then, we can write $A = QR$. This is the result of the QR-factorization. The matrix R is upper-triangular and Q is the unitary matrix. The matrix operation ‘ ’ stands for the matrix transposition.

Let us estimate the number $m(5)$ of multiplications in the QR-factorization of the 5×5 matrix A. The first 5-point DsiHT is applied to five columns of A. Then, the second 4-point DsiHT is applied to four columns of the 4×4 submatrix of A_1, and so on. Thus, considering the number of zero coefficients in the matrices of the DsiHTs, we obtain the following estimation:

$$m(5) = 5\mu(5) + 4\mu(4) + 3\mu(3) + 2\mu(2) = 5 \cdot 17 + 4 \cdot 12 + 3 \cdot 8 + 2 \cdot 4 = 165. \tag{35}$$

The number of rotations that perform the QR-factorization of the matrix is equal to $\alpha(N) = 4 + 3 + 2 + 1 = 10$. If use the Householder transformations in the QR-factorization of the same 5×5 matrix, the number of multiplications will be estimated by $5 \cdot 25 + 4 \cdot 16 + 3 \cdot 9 + 2 \cdot 4 = 224$.

Example 2. *Consider the following randomly generated 5×5 matrix:*

 doi:10.20944/preprints202512.1575.v1

$$A = \begin{bmatrix} 4 & 3 & 1 & 5 & -6 \\ 8 & 1 & -3 & 5 & 9 \\ -7 & 6 & 2 & 8 & 3 \\ 9 & 8 & 3 & -5 & 7 \\ 5 & 4 & -2 & 9 & 3 \end{bmatrix}, \qquad \det A \neq 0. \tag{36}$$

The QR-factorization of the matrix, $A \to R$, where the upper triangular matrix R is calculated as follows. The first 5-point DsiHT with fast path #4 is generated by the column $x = (4,8,-7,9,5)'$. Therefore, the first matrix in the factorization is equal to

$$H_5 = \begin{bmatrix} 0.2609 & 0.5219 & -0.4566 & 0.5871 & 0.3262 \\ -0.3312 & 0.4111 & 0.5796 & 0.4625 & -0.4140 \\ 0.4609 & 0 & 0.6749 & 0 & 0.5762 \\ 0 & -0.7474 & 0 & 0.6644 & 0 \\ -0.7809 & 0 & 0 & 0 & 0.6247 \end{bmatrix}, \qquad \det H_5 = 1. \tag{37}$$

At the next three stages of the matrix factorization, the matrices are composed by the matrices of the 4-,3-, and 2-point DsiHTs and are equal to

$$1 \oplus H_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.4817 & 0.7545 & 0.4454 & 0.0152 \\ 0 & -0.5541 & 0.6559 & -0.5124 & 0.0132 \\ 0 & -0.6789 & 0 & 0.7342 & 0 \\ 0 & 0 & -0.0202 & 0 & 0.9998 \end{bmatrix}, \qquad \det H_4 = 1, \tag{38}$$

and

$$I_2 \oplus H_3 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0.7060 & -0.4796 & 0.5211 \\ 0 & 0 & 0.3859 & 0.8775 & 0.2848 \\ 0 & 0 & -0.5939 & 0 & 0.8046 \end{bmatrix}, \quad I_3 \oplus H_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0.2077 & -0.9782 \\ 0 & 0 & 0 & 0.9782 & 0.2077 \end{bmatrix}. \tag{39}$$

The unitary matrix Q is calculated by Equation (34) and is equal to

$$Q = \begin{bmatrix} 0.2609 & 0.1764 & -0.1838 & 0.9304 & -0.0383 \\ 0.5219 & -0.1349 & 0.5066 & -0.0483 & -0.6712 \\ -0.4566 & 0.7885 & 0.2675 & 0.0186 & -0.3129 \\ 0.5871 & 0.5187 & -0.5046 & -0.3628 & -0.0025 \\ 0.3262 & 0.2448 & 0.6192 & 0.0121 & 0.6709 \end{bmatrix}, \quad \det Q = 1. \tag{40}$$

The triangular matrix is

$$R = Q'A = \begin{bmatrix} 15.3297 & 4.5663 & -1.1090 & 0.2609 & 6.8494 \\ 0 & 10.2542 & 3.2244 & 6.1251 & 4.4590 \\ 0 & 0 & -3.9209 & 11.8495 & 4.7902 \\ 0 & 0 & 0 & 6.4810 & -8.4648 \\ 0 & 0 & 0 & 0 & -4.5743 \end{bmatrix}. \tag{41}$$

Ten angles of rotations in these four DsiHTs are given in Table 5 (in degrees). This is an encoded table of the QR-factorization by the DsiHTs with path #4.

Table 5. Angles of rotations in the QR factorization of the 5×5 matrix A.

	ϑ_1	ϑ_2	ϑ_3	ϑ_4
H_5	$-51.3402°$	$47.5498°$	$-48.3665°$	$-51.7676°$
H_4	$-42.7596°$	$-1.1563°$	$-48.9986°$	
H_3	$-36.4314°$	$28.6623°$		
H_2	$78.0111°$			

We remind that in the QR-factorization, the triangularization $T: A \to R$ is accomplished by a unitary matrix Q. The original matrix is integer-valued, and it always possible to fulfil the triangularization by the integer matrices only, as shown below

 doi:10.20944/preprints202512.1575.v1

$$R_i = TA = \begin{bmatrix} 2 & 0 & 1 & 0 & 0 \\ -12 & 11 & 5 & 0 & -1 \\ -33 & 1 & -1 & 13 & 0 \\ 1 & 15 & 7 & 0 & -15 \\ 153 & 2681 & 1250 & 10 & -2680 \end{bmatrix} A = \begin{bmatrix} 1 & 12 & 4 & 18 & -9 \\ 0 & 1 & -33 & 26 & 183 \\ 0 & 0 & 1 & -233 & 295 \\ 0 & 0 & 0 & 1 & 105 \\ 0 & 0 & 0 & 0 & 18991 \end{bmatrix}. \quad (42)$$

The determinant of the matrix T is equal to $\det T = 1$, but this matrix is not unitary. To calculate the matrix A from this equation as $A = T^{-1}R_i$, the inverse of the matrix T^{-1} is required, which is not the transpose matrix, that is, $Q = T^{-1} \neq T'$.

The QR-factorization of the same matrix A by the Householder reflections results in the matrices

$$Q = \begin{bmatrix} -0.2609 & -0.1764 & -0.1838 & -0.9304 & 0.0383 \\ -0.5219 & 0.1349 & 0.5066 & -0.0483 & 0.6712 \\ 0.4566 & -0.7885 & 0.2675 & -0.0186 & 0.3129 \\ -0.5871 & -0.5187 & -0.5046 & 0.3628 & 0.0025 \\ -0.3262 & -0.2448 & 0.6192 & -0.0121 & -0.6709 \end{bmatrix}, \quad \det Q = 1, \quad (43)$$

and

$$R = \begin{bmatrix} -15.3297 & -4.5663 & 1.1090 & -0.2609 & -6.8494 \\ 0 & -10.2542 & -3.2244 & -6.1251 & -4.4590 \\ 0 & 0 & -3.9209 & 11.8495 & 4.7902 \\ 0 & 0 & 0 & -6.4810 & 8.4648 \\ 0 & 0 & 0 & 0 & 4.5743 \end{bmatrix}. \quad (44)$$

It can be noted that the coefficients of these matrices differ only in signs from the corresponding matrices of the DsiHT-based QR factorization, given in Equations (40) and (41). This example illustrates the advantage of the DsiHT-based method over the Householder reflection method. The results are almost the same, but the DsiHTs reduce the operations of multiplication by the number $224 - 165 = 59$, plus the encoded QR-factorization table is generated. Such a table can be used to create a unitary matrix 5×5.

The numbers of operations of multiplication for the QR-factorization by the Householder reflections and DsiHT with paths #1 and #4, are calculated as

$$m_H(N) = 2^3 + 3^3 + 4^3 + \cdots + N^3 = \frac{1}{4}N^2(N+1)^2 - 1, \quad (45)$$

$$m_{\#1,2}(N) = \sum_{n=2}^{N} n\,\mu_{\#1,2}(n) = \sum_{n=2}^{N} n\frac{1}{2}(n^2 + 3n - 2) = \frac{1}{2}\sum_{n=2}^{N}(n^3 + 3n^2 - 2n)$$
$$= \frac{1}{8}[N^2(N+1)^2 - 4] + \frac{1}{4}[N(N+1)(2N+1) - 6] - \frac{1}{2}(N+2)(N-1), \quad (46)$$

$$m_{\#3,4}(N) = \sum_{n=2}^{N} n\,\mu_{\#3,4}(n) = \sum_{k=2}^{N} n\left(n^2 - z_{\#3,4}(n)\right), \quad (47)$$

where $z_{\#3,4}(n) = z(n)$ is calculated by Equation (31). The values of these estimations for small values of $N = 3{:}14$ and for powers of two, 128, 256,512,1024, and 2048, are given in Table 6.

Table 6. Numbers of multiplication in the QR factorizations of the $N \times N$ matrix.

N	3	4	5	6	7	8	9	10	11	12	13	14
$m_H(N)$	35	99	224	440	783	1295	2024	3024	4355	6083	8280	11024
$m_{\#1}(N)$	32	84	179	335	573	917	1394	2034	2870	3938	5277	6929
$m_{\#4}(N)$	32	80	165	297	486	742	1084	1524	2074	2746	3552	4504

N	128	256	512	1024	2048
$m_H(N)$	6.81615×10^7	1.08215×10^9	1.7247×10^{10}	2.75415×10^{11}	4.40234×10^{12}
$m_{\#1}(N)$	3.51334×10^7	5.49478×10^8	8.6907×10^9	1.38245×10^{11}	2.20547×10^{12}
$m_{\#4}(N)$	5.35852×10^6	4.82302×10^7	4.2953×10^8	3.78943×10^9	3.31578×10^{10}

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 December 2025 doi:10.20944/preprints202512.1575.v1

One can note that the DsiHT-based method with the fast path #4 required a smaller number of multiplications when comparing with the path #1, as well as the Householder reflection-based method. The graphs of the functions $m_H(N)$ and $m_{\#1}(N)$ are shown in Figure 9 in part (a), when $N = 3{:}\,512$, and in part (b) when $N = 512{:}\,2048$.

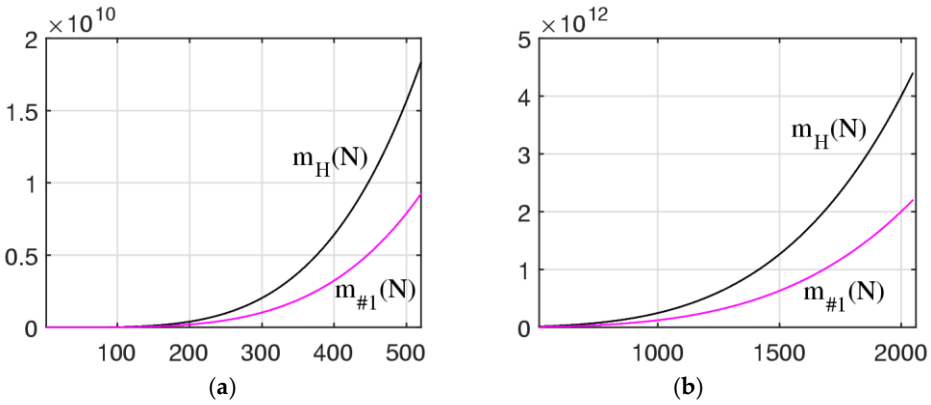


Figure 9. The curves of the functions $m_H(N)$ and $m_{\#1}(N)$ in the integer intervals (a) [2,512] and (b) [512,2048].

For comparison with the method of DsiHT with path #4, Figure 10 illustrates the ratios of functions $r_1(N) = m_H(N)/m_{\#4}(N)$ and $r_2(N) = m_{\#1}(N)/m_{\#4}(N)$ in the integer interval [2,2048]. These two curves are approximately equal to the lines with the slops $a_1 = m_H(2048)/m_{\#4}(2048) \approx 132.77$ and $a_2 = m_{\#1}(2048)/m_{\#4}(2048) \approx 66.51$.

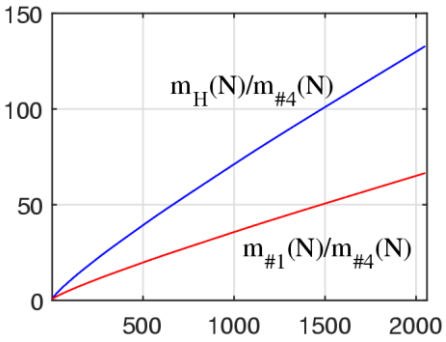


Figure 10. The curves of the functions (a) $r_1(N)$ and (b) $r_2(N)$.

5. Complex QsiHT

In this section, we describe the complex DsiHTs which can be used in QR-factorization of complex matrices. As shown in [32], there are different complex matrices 2×2 that can be used as the basis elements to compose the N-point complex DsiHT.

A. First, we consider the following 2×2 matrix:

$$M = \frac{1}{\sqrt{|x|^2 + |y|^2}} \begin{bmatrix} \bar{x} & \bar{y} \\ -y\dfrac{\bar{x}}{|x|} & |x| \end{bmatrix}. \tag{48}$$

Numbers x_0 and x_1 are complex. The matrix M is unitary and its determinant $\det M = \bar{x}/|x|$ and $|\det M| = 1$. The matrix product

$$M \begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{\sqrt{|x|^2 + |y|^2}} \begin{bmatrix} |x|^2 + |y|^2 \\ 0 \end{bmatrix} = \begin{bmatrix} \sqrt{|x|^2 + |y|^2} \\ 0 \end{bmatrix}. \tag{49}$$

It should be noted that such matrices are used in quantum computation for realization of complex multi-qubit operations [33,34]. Therefore, this matrix is decomposed by basic 2×2

 doi:10.20944/preprints202512.1575.v1

operations, or gates. For that, the polar form of the numbers is considered, that is, $x = |x|e^{i\varphi_0}$ and $y = |y|e^{i\varphi_1}$. Then, Equation (48) can be written as

$$M = \frac{1}{\sqrt{|x|^2 + |y|^2}} \begin{bmatrix} |x|e^{-i\varphi_0} & |y|e^{-i\varphi_1} \\ -|y|e^{i(\varphi_1-\varphi_0)} & |x| \end{bmatrix}. \tag{50}$$

This complex matrix can be encoded by four angles in the following way. First, we calculate the angle $\vartheta = -\operatorname{atan}(|y|/|x|)$ and then denote the cosine and sine coefficients

$$\cos\vartheta = \frac{|x|}{\sqrt{|x|^2 + |y|^2}} \quad \text{and} \quad \sin\vartheta = -\frac{|y|}{\sqrt{|x|^2 + |y|^2}}. \tag{51}$$

The matrix M can be written as

$$M = \begin{bmatrix} e^{-i\varphi_0}\cos\vartheta & -e^{-i\varphi_1}\sin\vartheta \\ e^{i(\varphi_1-\varphi_0)}\sin\vartheta & \cos\vartheta \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\varphi_0} \end{bmatrix} \begin{bmatrix} e^{-i\varphi_0}\cos\vartheta & -e^{-i\varphi_1}\sin\vartheta \\ e^{i\varphi_1}\sin\vartheta & e^{i\varphi_0}\cos\vartheta \end{bmatrix}, \tag{52}$$

or

$$M = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\varphi_0} \end{bmatrix} \begin{bmatrix} e^{-i(\varphi_0+\varphi_1)/2} & 0 \\ 0 & e^{i(\varphi_0+\varphi_1)/2} \end{bmatrix} \begin{bmatrix} \cos\vartheta & -\sin\vartheta \\ \sin\vartheta & \cos\vartheta \end{bmatrix} \begin{bmatrix} e^{-i(\varphi_0-\varphi_1)/2} & 0 \\ 0 & e^{i(\varphi_0-\varphi_1)/2} \end{bmatrix}. \tag{53}$$

The first matrix in this factorization is known as the phase shift, the 2nd and 4th ones are Z-rotations [35], and the 3rd matrix is the matrix of the elementary rotation. Thus, we can encode the matrix M by four angles as $\Phi = \{\varphi_0, \varphi_0 + \varphi_1, \vartheta, \varphi_0 - \varphi_1\}$. The QR-factorization of a complex matrix by such complex M matrices is described in detail in [32]. For this purpose, the concepts of the complex DsiHT with the strong and weak carriage-wheels are used.

Other 2×2 matrices also can be used, instead of the matrix M. We mention the known complex Givens rotation with the matrix [4]

$$G = \frac{1}{\sqrt{|x_0|^2 + |x_1|^2}} \begin{bmatrix} |x_0| & \frac{x_0}{|x_0|}\bar{x}_1 \\ -x_1\frac{\bar{x}_0}{|x_0|} & |x_0| \end{bmatrix}, \quad \det G = 1. \tag{54}$$

When applying the matrix G on the vector-generator $(x_0, x_1)'$, we obtain $x_0/|x_0|\left(\sqrt{|x_0|^2 + |x_1|^2}, 0\right)'$. Two coefficients of this matrix are real, and in the matrix M, only one coefficient is real. We can consider the complex matrix with all complex coefficients. For instance, the unitary matrix is defined as

$$T = \frac{\operatorname{sign}(\operatorname{Real}(x_0))}{\sqrt{|x_0|^2 + |x_1|^2}} \begin{bmatrix} \bar{x}_0 & \bar{x}_1 \\ -x_1 & x_0 \end{bmatrix}, \tag{55}$$

for which $T(x_0, x_1)' = \operatorname{sign}(\operatorname{Real}(x_0)\left(\sqrt{|x_0|^2 + |x_1|^2}, 0\right)'$. The complex vector (x_0, x_1) is rotated to the positive or negative direction of the real axis. The matrices G and T can be expressed by the matrix M as

$$G = \begin{bmatrix} \frac{x_0}{|x_0|} & 0 \\ 0 & 1 \end{bmatrix} M \quad \text{and} \quad T = \operatorname{sign}(\operatorname{Real}(x_0) \begin{bmatrix} 1 & 0 \\ 0 & \frac{|x_0|}{\bar{x}_0} \end{bmatrix} M, \qquad \text{if } x_0 \neq 0. \tag{56}$$

The comparison of QR-factorizations of a complex square matrix by the matrices M, G, and T, when using the weak and strong DsiHT, is given in [32] together with examples and MATLAB-based codes.

The following should be noted about the matrix M. The QR-factorization by this matrix can be calculated analytically without rotation angles and matrices. For simplicity of calculations, we consider the weak DsiHT. First, let the matrix A be real. The following notations for an input N-dimensional vector $\boldsymbol{z}$ and the vector generator $\boldsymbol{x}$ are used:

$$E_n(\boldsymbol{z}, \boldsymbol{x}) = z_0x_0 + z_1x_1 + \cdots + z_{n-1}x_{n-1}, \quad n = 1{:}N, \tag{57}$$

and

 doi:10.20944/preprints202512.1575.v1

$$[E_n(\boldsymbol{x})]^2 = E_n(\boldsymbol{x},\boldsymbol{x}) = x_0^2 + x_1^2 + \cdots + x_{n-1}^2, \tag{58}$$

for the partial energies $E_n(\boldsymbol{x})$ of the signal generator. The DsiHT of the input vector $\boldsymbol{z} = (z_0, z_1, \ldots, z_{N-1})$, that is, $H_{N,\#1}\boldsymbol{z} = (z_0^{(N-1)}, z_1^{(1)}, z_2^{(1)}, \ldots, z_{N-1}^{(1)})$ can be calculated by the correlation data, as follows:

$$z_0^{(N-1)} = E_N(\boldsymbol{z},\boldsymbol{x})/E_N(\boldsymbol{x}), \tag{59}$$

$$z_n^{(1)} = \frac{E_n(\boldsymbol{x},\boldsymbol{x})z_n - E_n(\boldsymbol{z},\boldsymbol{x})x_n}{E_{n+1}(\boldsymbol{x})E_n(\boldsymbol{x})}, \qquad n = 1{:}\,(N-1). \tag{60}$$

For a given generator $\boldsymbol{x}$, all values of $E_n(\boldsymbol{x},\boldsymbol{x})$ and $E_{n-1}(\boldsymbol{x})E_n(\boldsymbol{x})$, can be calculated in advance. The coefficients $h_{n,m}, n, m = 0{:}\,(N-1)$, of the matrix $H_{N,\#1}$ can be obtained from Equations (57)–(60). For that, the standard procedure is used. The m-th column of this matrix is the DsiHT of the unit vector $\boldsymbol{e}_m = (0, 0, \ldots, 1, \ldots, 0)'$, with 1 on the m-th position. Therefore, all coefficients can be calculated using the formulas

$$h_{0,m} = \frac{E_N(\boldsymbol{e}_m,\boldsymbol{x})}{E_N(\boldsymbol{x})}, \quad h_{n,m} = \frac{[E_n(\boldsymbol{x})]^2(\boldsymbol{e}_m)_n - E_n(\boldsymbol{e}_m,x)x_n}{E_{n+1}(\boldsymbol{x})E_n(\boldsymbol{x})}, \quad n = 1{:}\,(N-1). \tag{61}$$

The diagonal coefficients of this matrix $h_{n,n} = E_n(\boldsymbol{x})/E_{n+1}(\boldsymbol{x})$ represent the ratios of the energies of the first n components of the signal to the $(n+1)$ components.

When the matrix A is complex, the complex DsiHT can be calculated by using the similar formulas. The partial cross-correlations of the complex input $\boldsymbol{z}$ with the complex generator $\boldsymbol{x}$ are calculated by

$$E_n(\boldsymbol{z},\boldsymbol{x}) = z_0\overline{x}_0 + z_1\overline{x}_1 + \cdots + z_{n-1}\overline{x}_{n-1}, \; n = 1{:}\,N, \tag{62}$$

and the partial energies $E_n(\boldsymbol{x})$ of the generator are calculated by

$$[E_n(\boldsymbol{x})]^2 = E_n(\boldsymbol{x},\boldsymbol{x}) = |x_0|^2 + |x_1|^2 + \cdots + |x_{n-1}|^2. \tag{63}$$

The above matrices M, G, and T can be used for calculating the DsiHTs with fast paths #3 and #4. However, we consider another and more effective way to perform the basic operation $(x, y) \to (z, 0)$ on complex data.

B. We introduce a new matrix instead of the above matrix M. First, we remove the phases from the complex numbers x and y, as $(x, y) \to (|x|, |y|)$. In the matrix form, this operation can be written as

$$\begin{bmatrix} e^{-i\varphi_0} & 0 \\ 0 & e^{-i\varphi_1} \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} e^{-i\varphi_0} & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & e^{-i\varphi_1} \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} |x| \\ |y| \end{bmatrix}. \tag{64}$$

Then, the elementary rotation can be used on the real vector,

$$T\begin{bmatrix} |x| \\ |y| \end{bmatrix} = \begin{bmatrix} \cos\vartheta & -\sin\vartheta \\ \sin\vartheta & \cos\vartheta \end{bmatrix}\begin{bmatrix} |x| \\ |y| \end{bmatrix} = \begin{bmatrix} \sqrt{|x|^2 + |y|^2} \\ 0 \end{bmatrix}. \tag{65}$$

Here, the rotation angle is calculated as $\vartheta = -\arctan(|y|/|x|)$. If $x = 0$, then $\vartheta = -\pi/2$. We obtain the following complex-to-real transformation:

$$A{:}\,(\boldsymbol{x},\boldsymbol{y}) \to \left(\sqrt{|x|^2 + |y|^2}, 0\right), \tag{66}$$

with the matrix

$$A = \begin{bmatrix} \cos\vartheta & -\sin\vartheta \\ \sin\vartheta & \cos\vartheta \end{bmatrix}\begin{bmatrix} e^{-i\varphi_0} & 0 \\ 0 & e^{-i\varphi_1} \end{bmatrix}. \tag{67}$$

This matrix is simple when compared to the matrix M and it can be encoded by three angles $\Psi = \{\varphi_0, \varphi_1, \vartheta\}$. We also can write $A = A(\varphi_0, \varphi_1, \vartheta)$. If the input x and y are real, we prefer to use the original matrix of rotation, which is defined in Equation (1). Then, the matrix will be encoded as $\Psi = \{0,0,\vartheta\}$. The complex N-point DsiHT with the basic matrix A is a straightforward extension of the real N-point DsiHT.

 doi:10.20944/preprints202512.1575.v1

Example 3. *Let us consider the complex 5-point signal*

$$\boldsymbol{x} = (x_0, x_1, x_2, x_3, x_4) = (1+i, -2+3i, 5+4i, 3+i, 4-2i), \quad \|\boldsymbol{x}\| = \sqrt{86}. \tag{68}$$

The matrix of the 5-point DsiHT generated by $\boldsymbol{x}$ and path #4 has the following complex unitary 4×4 matrix:

$$H_5 = \begin{bmatrix} 0.1078 & -0.2157 & 0.5392 & 0.3235 & 0.4313 \\ -0.0652 & -0.3569 & -0.3258 & 0.5354 & -0.2606 \\ -0.1720 & 0 & 0.4614 & 0 & -0.6880 \\ 0 & 0.3658 & 0 & 0.7132 & 0 \\ -0.6742 & 0 & 0 & 0 & 0.2697 \end{bmatrix}$$
$$+i\begin{bmatrix} -0.1078 & -0.3235 & -0.4313 & -0.1078 & 0.2157 \\ 0.0652 & -0.5354 & 0.2606 & -0.1785 & -0.1303 \\ 0.1720 & 0 & -0.3692 & 0 & -0.3440 \\ 0 & 0.5486 & 0 & -0.2377 & 0 \\ 0.6742 & 0 & 0 & 0 & 0.1348 \end{bmatrix}. \tag{69}$$

This matrix has eight zero coefficients, as in the case of real matrices. The determinant of the matrix is equal to $\det H_5 = -0.9443 + 0.3292i$ and $|\det H_5| = 1$. Up to the constant $0.1078 = 1/\sqrt{86}$, the first row of the matrix is the generator $\boldsymbol{x}$, that is, it is equal to $\boldsymbol{x}/\|\boldsymbol{x}\|$. The diagram of this transformation is shown in Figure 11 and it looks like what is shown in Figure 7 in part (b). Only, the butterflies with rotations $T_k, k = 1{:}4,$ are changed by the complex operations $A_k = A(\varphi_{0;k}, \varphi_{1;k}, \vartheta_k)$.

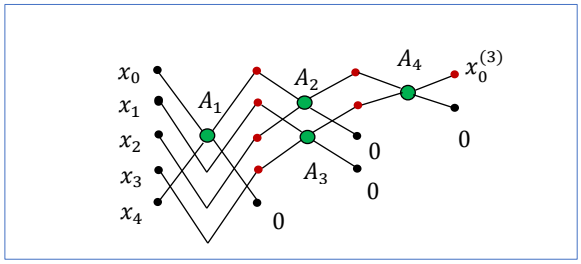


Figure 11. The diagram for calculating the 5-point complex DsiHT with path #4.

All angles of the complex operations $A_k, k = 1{:}4,$ in this DsiHT are given in Table 7.

Table 7. The angles of the 5-point complex DsiHT.

	$\varphi_{0;k}$	$\varphi_{1;k}$	ϑ_k	k
A_1	45°	−26.5651°	−72.4516°	1
A_2	0°	38.6598°	−53.7765°	2
A_3	123.6901°	18.4349°	−41.2526°	3
A_4	0°	0°	−31.1411°	4

Now, we consider the example of the QR-factorization of a complex matrix by the DsiHT with path #4.

Example 4. *Consider the following 5×5 complex matrix:*

$$A = \begin{bmatrix} 4 & 3 & 1 & 5 & -6 \\ 8 & 1 & -3 & 5 & 9 \\ -7 & 6 & 2 & 8 & 3 \\ 9 & 8 & 3 & -5 & 7 \\ 5 & 4 & -2 & 9 & 3 \end{bmatrix} + i\begin{bmatrix} -1 & 1 & 3 & 5 & 4 \\ 3 & 5 & -1 & 1 & -1 \\ 2 & 3 & 4 & -2 & 4 \\ -2 & 2 & 3 & -2 & -1 \\ -2 & -1 & 5 & -1 & 5 \end{bmatrix}, \quad |\det A| > 117710. \tag{70}$$

The QR-factorization of this matrix, $H_2H_3H_4H_5{:}\, A \to R,$ by four complex DsiHTs, each with path #4, results in the following matrices:

 doi:10.20944/preprints202512.1575.v1

$$Q_H = Q_{H;\#4} = \begin{bmatrix} 0.2495 & 0.1281 & 0.1843 & 0.2960 & 0.1733 \\ 0.4990 & -0.1047 & -0.0140 & 0.4973 & -0.5008 \\ -0.4366 & 0.7785 & 0.0203 & 0.1934 & -0.2135 \\ 0.5614 & 0.4034 & 0.4464 & -0.3933 & 0.0999 \\ 0.3119 & 0.1699 & -0.4737 & 0.2512 & -0.0346 \end{bmatrix}$$
$$+i\begin{bmatrix} -0.0624 & 0.0528 & 0.2389 & 0.5986 & 0.5896 \\ 0.1871 & 0.2200 & -0.3370 & -0.2036 & 0.0567 \\ 0.1248 & 0.3216 & -0.0005 & -0.0175 & -0.0227 \\ -0.1248 & 0.0891 & -0.1497 & 0.0693 & -0.3277 \\ -0.1248 & -0.1109 & 0.5905 & 0.0709 & -0.4512 \end{bmatrix} \quad (71)$$

and (upper triangular)

$$R_H = R_{H;\#4} = \begin{bmatrix} 16.0312 & 5.4893 & -1.9337 & 0.2495 & 6.1131 \\ 0 & 11.2651 & 3.8074 & 5.6319 & 4.5929 \\ 0 & 0 & 6.1090 & -4.9145 & 4.9253 \\ 0 & 0 & 0 & 12.3561 & 4.0932 \\ 0 & 0 & 0 & 0 & -5.3096 \end{bmatrix}$$
$$+i\begin{bmatrix} 0 & 2.9942 & 2.2456 & 0.0624 & -1.4347 \\ 0 & 0 & 5.1383 & -4.4913 & 1.2587 \\ 0 & 0 & 0 & -5.1204 & 1.7614 \\ 0 & 0 & 0 & 0 & 7.8876 \\ 0 & 0 & 0 & 0 & 6.8097 \end{bmatrix}. \quad (72)$$

We need to mention that the coefficients in the diagonal of the upper triangular matrix R_H are real, except the last coefficient. Therefore, the imaginary part of the matrix has four more zeros than the real part. The angles of all ten complex operations A_k in this QR-factorization are given in Table 8.

Table 8. The angles of the DsiHT-based QR-factorization of A.

		$\varphi_{0;k}$	$\varphi_{1;k}$	ϑ_k	k
H_5	A_1	$-14.0362°$	$-21.8014°$	$-52.5608°$	1
	A_2	$0°$	$164.0546°$	$-47.0273°$	2
	A_3	$20.5560°$	$-12.5288°$	$-47.1779°$	3
	A_4	$0°$	$0°$	$-51.6359°$	4
H_4	A_1	$48.3521°$	-12.4259°	$-23.6165°$	1
	A_2	$-148.9744°$	$-70.1077°$	$-7.5599°$	2
	A_3	$0°$	$0°$	$-46.4066°$	3
H_3	A_1	$106.5349°$	$-107.2347°$	-42.6405°	1
	A_2	$0°$	$54.6890°$	$-36.4034°$	2
H_2	A_1	$97.5470°$	$88.2841°$	$-54.3345°$	1

The unitary matrix is equal to $Q_H = (H_2H_3H_4H_5)'$. Table 7 can be considered as an encoded table of angles for Q_H. Any such encoded table, namely the set of 30 angles, can be used to generate a complex unitary 5×5 matrix. In the general case of $N > 2$, such encoded tables of angles for $(N-1)$ DsiHTs with path #4 can be used to generate a complex unitary $N \times N$ matrix. The number of angles in the encoded table is equal to $3/2N(N-1)$.

Now, we analyze the MATLAB version of QR-factorization of the same matrix, $A = Q_M R_M$, which is calculated by the function 'qr.m,' as '[Q_M,R_M]=qr(A).' The method of the Householder transformations is programmed in this function. Many coefficients of these two matrices are equal up to the sign to the coefficients of the corresponding matrices Q_H and R_H. Therefore, we consider the pointwise ratio of the matrices,

$$Real(Q_H)./Real(Q_M) = \begin{bmatrix} -1 & 1 & -1 & 1 & 0.3032 \\ -1 & 1 & -1 & 1 & 1.9027 \\ -1 & 1 & -1 & 1 & 1.4308 \\ -1 & 1 & -1 & 1 & -0.5074 \\ -1 & 1 & -1 & 1 & 0.0918 \end{bmatrix}, \quad (73)$$

 doi:10.20944/preprints202512.1575.v1

$$Imag(Q_H)./Imag(Q_M) = \begin{bmatrix} -1 & 1 & -1 & 1 & 2.6100 \\ -1 & 1 & -1 & 1 & 0.1320 \\ -1 & 1 & -1 & 1 & -0.1473 \\ -1 & 1 & -1 & 1 & 1.1690 \\ -1 & 1 & -1 & 1 & 1.8038 \end{bmatrix}, \quad (74)$$

and

$$Real(R_H)./Real(R_M) = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ * & 1 & 1 & 1 & 1 \\ * & * & -1 & -1 & -1 \\ * & * & * & 1 & 1 \\ * & * & * & * & 0.6149 \end{bmatrix}, \quad (75)$$

$$Imag(R_H)./Imag(R_M) = \begin{bmatrix} * & -1 & -1 & -1 & -1 \\ * & * & 1 & 1 & 1 \\ * & * & * & -1 & -1 \\ * & * & * & * & 1 \\ * & * & * & * & 6.8097/0 \end{bmatrix}. \quad (76)$$

Here, the symbol '*' is used at points where the coefficients of the matrices are zero (as 0/0). The last coefficients of the triangular matrices $R_H(5,5) = -5.3096 + 6.8097i$ and $R_M(5,5) = -8.6350$. The major difference of these matrices is in the coefficients of the last columns.

Although we are writing $A = Q_H R_H$ and $A = Q_M R_M$, there are small errors in calculation of the matrix A by Q and R matrices. To compare the results of the QR-factorization, the precision of computation has been estimated by the 2-norms of the matrix $\Delta_H = (A - Q_H R_H)$ and matrix $\Delta_M = (A - Q_M R_M)$, by using the MATLAB function "norm.m." The results of the calculation are

$$\|\Delta_M\| = 1.3688 \times 10^{-14}, \quad \|\Delta_H\| = 0.4956 \times 10^{-14}, \text{ and } \quad \|\Delta_M\|/\|\Delta_H\| = 2.7621. \quad (77)$$

Relative to the 2-norm, the error of the QR-factorization by the DsiHTs with fast path #4 is more than two times less than the error when using the method of the Householder transformations.

As mentioned above, the fast paths, like paths #3 and #4, allow us to decompose the matrix A by using matrices with more zero coefficients than when using the paths #1 and #2. It means that fewer operations of multiplication and addition will be used, and results will be more accurate. That can be verified in this example as well. For instance, we consider the QR-factorization by DsiHTs with path #1 (that is, the DsiHTs with the weak wheel-carriage). The calculations by the analytical Equation (56) result in the following complex matrices:

$$Q_{H;\#1} = \begin{bmatrix} \circ & \circ & \circ & \circ & -0.6012 \\ \circ & \circ & \circ & \circ & -0.0183 \\ \circ & \circ & \circ & \circ & 0.0390 \\ \circ & \circ & \circ & \circ & 0.3191 \\ \circ & \circ & \circ & \circ & 0.4525 \end{bmatrix} + i \begin{bmatrix} \circ & \circ & \circ & \circ & 0.1277 \\ \circ & \circ & \circ & \circ & -0.5037 \\ \circ & \circ & \circ & \circ & -0.2111 \\ \circ & \circ & \circ & \circ & 0.1247 \\ \circ & \circ & \circ & \circ & 0 \end{bmatrix}, \quad (78)$$

$$R_{H;\#1} = \begin{bmatrix} 0 & \circ & \circ & \circ & \circ \\ 0 & \circ & \circ & \circ & \circ \\ 0 & 0 & \circ & \circ & \circ \\ 0 & 0 & 0 & \circ & \circ \\ 0 & 0 & 0 & 0 & 7.1959 \end{bmatrix} + i \begin{bmatrix} 0 & \circ & \circ & \circ & \circ \\ 0 & 0 & \circ & \circ & \circ \\ 0 & 0 & 0 & \circ & \circ \\ 0 & 0 & 0 & 0 & \circ \\ 0 & 0 & 0 & 0 & 4.7732 \end{bmatrix}. \quad (79)$$

Here, only those coefficients in the matrices $Q_{H;\#1}$ and $R_{H;\#1}$ are given that differ from the coefficients of the corresponding matrices $Q_{H;\#4}$ and $R_{H;\#4}$. All other coefficients are equal and marked by '°.' Note that all coefficients in the above matrices are given with an accuracy of up to 4 decimal places.

The results of the calculation of the 2-norm o the matrix $\Delta_{H;\#1} = A - Q_{H;\#1} R_{H;\#1}$ are the following:

$$\|\Delta_{H;\#1}\| = 1.1803 \times 10^{-14}, \quad \|\Delta_M\|/\|\Delta_{H;\#1}\| = 1.1597. \quad (80)$$

Relative to the 2-norm, the error of the QR-factorization by the DsiHTs with fast path #1 is smaller than the error when using the Householder transformations. However, this 2-norm is larger than the 2-norm $\|\Delta_{H;\#4}\|$ of the factorization by the fast paths. Namely, $\|\Delta_{H;\#1}\|/\|\Delta_{H;\#4}\| = 2.3816$.

 doi:10.20944/preprints202512.1575.v1

In conclusion, we consider the example of a complex matrix of a large size, and we compare the results of the QR-factorization by the DsiHTs with path #4 with the MATLAB version of the factorization.

Example 5. *The 256×256 complex matrix* A *is composed by two real images, as shown in Figure 12 part (a). The real part of this matrix is the 'cameramen.tif' image of 256×256 pixels. The imaginary part of this matrix represents the grayscale 'peppers.tiff' image of 512×* 512 *pixels after down-sampling to the size of 256×256 pixels. These two images are taken from the database by address: http://sipi.usc.edu/database. The real and imaginary parts of this complex matrix are integer-valued non-negative matrices. The result of the DsiHT-based QR-factorization of this 256×256 complex image* A *is shown in part (b). Visually it is difficult to see the difference between these two images in this figure. The difference matrix* Δ_H *has values in the small interval* $10^{-12} \times [-0.7105, 0.5684]$ *and is shown in part (c) as the image after scaling, by using the MATLAB function 'imagesc.m.' The 2-norm of the difference matrix* $\Delta_H = (A - Q_H R_H)$ *is equal to* $\|\Delta_H\| \approx 0.2054 \times 10^{-10}$.

(**a**) (**b**) (**c**)

Figure 12. (a) The original complex image A, (b) the complex image $Q_H R_H$, and (c) the difference of images.

Figure 13 shows the image of the unitary matrix Q_H together with the image of the triangular matrix R_H. Since the value of the matrix Q_H are small, this matrix is displayed in an absolute scale with the factor of 2000. The matrix of R_H is also displayed in the absolute scale with the factor of 3.

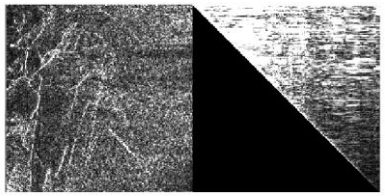


Figure 13. The image of the matrix $2000 \times |Q_H|$ and the image of $3 \times |R_H|$.

For comparison, Figure 14 in part (a)shows the result $Q_M R_M$ obtained when using MATLAB code 'qr(A).' The 2-norm of the difference matrix $\Delta_M = (A - Q_M R_M)$ is equal to $\|\Delta_M\| \approx 0.25344 \times 10^{-10}$. This 2-norm is 1.2337 times larger than the 2-norm $\|\Delta_H\|$. Since the original images are integer-valued, the results $Q_H R_H$ and $Q_M R_M$ can be rounded before displaying them. The rounding of the matrix $Q_M R_M$ is shown in Figure 14 in part (b). There is a zero error between the rounded matrix $Q_H R_H$ and the matrix A. The same is true for the rounded matrix $Q_M R_M$.

(**a**) (**b**)

Figure 14. (a) The complex image $Q_M R_M$ and (b) the rounded image of $Q_H R_H$.

 doi:10.20944/preprints202512.1575.v1

6. Conclusions

The discrete signal-induced heap transform-based method of the QR-factorization of a square matrix has different realizations. This is explained by the presence of different paths when performing the DsiHTs. Our goal is to show the fast paths which lead to effective calculation of the QR-factorization. The real and complex matrices are considered. Examples of the QR-factorization are given and described in detail for the 4×4, 5×5, and 8×8 real and complex matrices and compared with the Householder transform-based QR-factorization, which is used in MATLAB. The illustrative example with a complex matrix of size 256×256 is also given. Different paths of the DsiHTs were considered together with the traditional path, which we call path #1. The special attention is given to the fast paths, namely path #4. Probably it is the one of fast paths that allow to have the sparsest matrices of the DsiHTs in the QR-factorization and therefore to reduce the number of operations in calculations. We also assume that for complex matrices, the 2×2 matrix described in Equation (67), is the simplest matrix (the complex matrix of rotations) that can be used for effective calculation of the QR-factorization.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Author codes based on MATLAB are available on the web page https://ceid.utsa.edu/agrigoryan/codes/

Conflicts of Interest: The author declares no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DsiHT	Discrete signal-induced Heap transform
QR	QR factorization of the matrix
BP	Bit plane

References

1. Horn, R.A.; Charles, R.J.; *Matrix Analysis*. Cambridge University Press, **1985**.
2. Scott, D.T., Bryce G.R.; Allen, D.M. (1985). Orthogonalization-triangularization methods in statistical computations. Amer. Statist. 39, 128-135.
3. Coleman, T.F.; Van Loan, C.F. Handbook for Matrix Computations. SIAM, Philadelphia, PA, **1988**.
4. Golub, G.H.; Van Loan, C.F. *Matrix Computations*, 3rd edition, Johns Hopkins, **1996**.
5. Goodall, C.R. Computation using the QR decomposition, Ch 13, pp. 467-508 (in C. R. Rao, ed., Handbook of Statistics, Vol. 9, Elsevier Science Publishers B.V. **1993**).
6. Trefethen, L.N.; Bau, David III. Numerical linear algebra. Philadelphia, PA: Society for Industrial and Applied Mathematics, **1997**.
7. Moon, T.K.; Stirling, W.C. Mathematical Methods and Algorithms for Signal Processing. NJ: Prentice Hall, 2000.
8. Xiao, L.; He, Y.; Li, Y.; Dai, J. Design and analysis of two nonlinear ZNN models for matrix LR and QR factorization with application to 3-D moving target location, *IEEE Transactions on Industrial Informatics*, vol. 19, no. 6, pp. 7424-7434, June **2023**, doi: 10.1109/TII.2022.3210038.
9. Alizadeh, M.; Sajedi, H.; Babaali, B. Image watermarking by Q Learning and matrix factorization, *2020 International Conference on Machine Vision and Image Processing (MVIP)*, Iran, **2020**, pp. 1-7, doi: 10.1109/MVIP49855.2020.9116871.
10. Hai, N.T.; Thanh, T.M. Robust image watermarking algorithm integrating QR and singular value decomposition in the discrete wavelet transform domain, *2025 2nd International Conference On Cryptography And Information Security (VCRIS)*, Hanoi, Vietnam, **2025**, pp. 1-6, doi: 10.1109/VCRIS68011.2025.11250561.

11. Lu, X.; Li, B.; Zeng, J. Radar-embedded communication waveform design based on QR decomposition, *2024 4th International Symposium on Computer Technology and Information Science (ISCTIS)*, Xi'an, China, **2024**, pp. 841-848, doi: 10.1109/ISCTIS63324.2024.10698827.
12. Sokolovskiy, V.; Tyapkin, V.N.; Veisov, E.A.; Fateev, Y.L. The pipelined QR decomposition hardware architecture based on Givens rotation CORDIC algorithm, *2019 International Siberian Conference on Control and Communications (SIBCON)*, Tomsk, Russia, **2019**, pp. 1-4, doi: 10.1109/SIBCON.2019.8729615.
13. Saraf, N.; Bemporad, A. A bounded-variable least-squares solver based on stable QR updates," in *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 1242-1247, March **2020**, doi: 10.1109/TAC.2019.2925501.
14. Strang, G. Linear Algebra and Learning from Data (1st ed.). Wellesley: Wellesley Cambridge Press. p. 143, **2019**.
15. Subramani, C.; Kuppili, V.; Keshavamurthy, B.N.; Prasad, R.; Jagannath, K.; Prashanth, G.R. The least square QR method improves extreme learning machine, *2023 IEEE World Conference on Applied Intelligence and Computing (AIC)*, Sonbhadra, India, **2023**, pp. 1-6, doi: 10.1109/AIC57670.2023.10263978.
16. Guerrero-García, P.; Hendrix, E.M.T. Experiments with active-set LP algorithms allowing basis deficiency. Computers **2023**, 12, 3. https://doi.org/10.3390/computers12010003
17. Zhang, X.M.; Li, T.; Yuan, X. Quantum state preparation with optimal circuit depth: Implementations and applications, *Physical Review Letters*, **129** (23), **2022**.
18. Vartiainen, J.J.; Möttönen, M.; Salomaa, M.M. Efficient decomposition of quantum gates, *Phys. Rev. Lett.*, vol. 92, no. 17, p. 177902, Apr. **2004**.
19. Mikko M.; Juha, J. Vartiainen, Decompositions of general quantum gates, **2005**. arXiv:quant-ph/0504100v1
20. Plesch, M.; Brukner, C. Quantum state preparation with universal gate decompositions. arXiv 2010, arXiv:1003.5760v2.
21. A. Björck, Solving linear least squares problems by Gram-Schmidt orthogonalization, BIT **7 (1967)** 1-21.
22. P. Alonso, J.M. Peña, M.L. Serrano, *QR decomposition of almost strictly sign regular matrices*. *Journal of Computational and Applied Mathematics*, **318**, pp. 646-657, **2017**.
23. Rutishause, H. *Simultaneous Iteration Method for Symmetric Matrices*. In: Bauer F.L. (eds) Linear Algebra. Handbook for Automatic Computation, **186** (**2017**). Springer, Berlin, Heidelberg.
24. Householder, A.S. Unitary triangulation of a nonsymmetric matrix. *Journal ACM*, **5** (4), pp. 339-342, **1958.**
25. Hsiao. Shen-Fu; Delosme, J.-M. Householder CORDIC algorithms, *IEEE Transactions on Computers*, vol. 44, no. 8, pp. 990-1001, **1995**.
26. Ying, L.; Musheng, W.; Fengxia, Z.; Jianli, Z. Real structure-preserving algorithms of Householder based transformations for quaternion matrices, *Journal of Computational and Applied Mathematics*, **305** (2016), 82-91.
27. Businger, P.; Golub, G.H. *Linear Least Squares Solutions by Householder Transformations*. In: Bauer F.L. (eds) Linear Algebra. Handbook for Automatic Computation, **2** (**2017**). Springer, Berlin, Heidelberg.
28. Bindel, D.; Demmel, J.; Kahan, W.; Marques O. On computing Givens rotations reliably and efficiently. *LAPACK Working Note 148*, University of Tennessee, UT-CS-00-449, **2001**.
29. Demmel, J.; Grigori, L.; Hoemmen, M.; Langou, J. Communication-optimal parallel and sequential QR and LU factorizations, *SIAM J. Sci. Comp*., **34** (1), pp. 206-239, **2012**.
30. Weslley S Pereira Ali Lotfi Julien Langou, Numerical analysis of Givens rotation, arXiv preprint arXiv:2211.04010, **2022**
31. Grigoryan, A.M. New method of Givens rotations for triangularization of square matrices, *Journal of Advances in Linear Algebra & Matrix Theory (ALAMT)*, 4 (2), pp. 65-78, **2014**, doi: 10.4236/alamt.2014.42004
32. Grigoryan, A.M. Effective methods of QR-decompositions of square complex matrices by fast discrete signal-induced heap transforms, vol. 12, no. 4, pp. 87-110, *Advances in Linear Algebra & Matrix Theory (ALAMT)*, Oct. 26, **2023**, doi: 10.4236/alamt.2022.124005
33. Grigoryan, A.M. New permutation-free quantum circuits for implementing 3- and 4-qubit unitary operations, *Information* **2025**, 16, 621, p. 33. https://doi.org/10.3390/info16070621
34. Grigoryan, A.M.; Gomez, A.A.; Agaian, S.S. A novel approach to state-to-state transformation in quantum computing, *Information* **2025**, 16, 689., p. 31. https://www.mdpi.com/2078-2489/16/8/689
35. Nielsen, M.A.; Chuang, I.L. Quantum Computation and Quantum Information; Cambridge University Press: Cambridge, U.K., **2000**.

 doi:10.20944/preprints202512.1575.v1