

# A Low-Complexity Approach to Computation of the Discrete Fractional Fourier Transform

Dorota Majorkowska-Mech<sup>1</sup> · Aleksandr Cariow<sup>1</sup>

Received: 30 December 2015 / Revised: 17 January 2017 / Accepted: 19 January 2017 /  
Published online: 4 February 2017  
© The Author(s) 2017. This article is published with open access at Springerlink.com

**Abstract** This paper proposes an effective approach to the computation of the discrete fractional Fourier transform for an input vector of any length  $N$ . This approach uses specific structural properties of the discrete fractional Fourier transformation matrix. Thanks to these properties, the fractional Fourier transformation matrix can be decomposed into a sum of three or two matrices, one of which is a dense matrix, and the rest of the matrix components are sparse matrices. The aforementioned dense matrix has unique structural properties that allow advantageous factorization. This factorization is the main contributor to the reduction in the overall computational complexity of the discrete fractional Fourier transform computation. The remaining calculations do not contribute significantly to the total amount of computation. Thus, the proposed approach allows to reduce the number of arithmetic operations when calculating the discrete fractional Fourier transform.

**Keywords** Discrete linear transforms · Discrete fractional Fourier transform · Eigenvalue decomposition

**Mathematics Subject Classification** MSC 15A04 · MSC 15A18 · MSC 65Y20

---

✉ Dorota Majorkowska-Mech  
dmajorkowska@wi.zut.edu.pl  
Aleksandr Cariow  
acariow@wi.zut.edu.pl

<sup>1</sup> Faculty of Computer Science and Information Technology, West Pomeranian University of Technology, Szczecin, Poland

## 1 Introduction

Many discrete fractional transforms, such as discrete fractional cosine and sine transforms [22], discrete fractional random transform [12], discrete fractional Hartley transform [23], discrete fractional Hilbert transform [21], discrete fractional Hadamard transform [20], and discrete fractional Fourier transform [18] have been found very useful for signal processing [27], digital watermarking [5], image encryption [7, 10, 15], image and video processing [9]. Among other transformations, the discrete fractional Fourier transform is perhaps most frequently used. To date, a number of effective algorithms for various discrete fractional transforms have been developed [2, 16, 25]. Fractional Fourier transform (FRFT) is a generalization of the ordinary Fourier transform (FT) with one fractional parameter. This concept was initially introduced by Wiener [26]. The FRFT was recognized as a transform method by mathematicians in 1980 after the work of Namias [14] in which the FRFT was introduced as a fractional power of the ordinary FT operator. To compute the FRFT by using digital techniques a discrete version of this transform was needed. It initiated the work of defining a discrete version of FRFT (DFRFT). Existing approaches to the definition of the discrete Fourier transform can be categorized into three major types. The first approach is represented by direct sampling of the FRFT [16, 17]. It is the least complicated approach and there are quite a few different algorithms that have been developed for the computation of this type of DFRFT. But these discrete realizations could lose many important properties of the FRFT such as unitarity, reversibility, additivity; therefore, its applications are very limited. This approach is still used and developed [11]. The second approach relies on a linear combination of ordinary Fourier operators raised to different powers [4, 24]. However, as emphasized in [3], these realizations often produce a result that does not match the result of the continuous FRFT. In other words, it is not a discrete version of the continuous transform. The third approach is based on the idea of eigenvalue decomposition [3, 18, 19]. This type of DFRFT possesses all essential properties which are posed as requirements for DFRFT such as unitarity, additivity, reduction to discrete Fourier transform (when the power is equal to unity), approximation of the continuous FRFT. However, the eigenvalue decomposition approach cannot be written in a closed form and may be computationally costly. The authors are interested in this type of DFRFT because currently there are no fast algorithms created for its realization. The work [22] described the method to reduce the computational load of the DFRFT by one half. In that work authors defined the discrete fractional cosine transform (DFRCT), which is the fractional version of the first type discrete cosine transform (DCT-I), and the discrete fractional sine transform (DFRST)—the fractional version of the first type discrete sine transform (DST-I). They also discovered the relationships between the eigenvectors of the DFRCT and DFRFT, as well as between the eigenvectors of DFRST and DFRFT matrices. The authors proved that the eigenvectors of the DFRCT matrix of size  $N$  can be easily obtained from the even eigenvectors of the DFRFT matrix of size  $2N - 2$ . Similarly, they showed that the eigenvectors of the DFRST matrix of size  $N$  can be obtained from the odd eigenvectors of the DFRFT matrix of size  $2N + 2$ . These results allowed them to demonstrate that the DFRFT transform of an even size  $N$  can be obtained through splitting the input signal into the even and odd parts, proper truncation of

each part and then computing DFRCT of size  $N/2 + 1$  from the even part and DFRST of size  $N/2 - 1$  from the odd part. After this operations are performed, the output signals should be extended properly and in the end added (before the addition one of them should be multiplied by the factor determined). This method is quite elegant and allows to reduce the computational cost by about one half, but it works only for signals of an even length  $N$ .

The work [8] presented a comparative analysis of the most famous algorithms for the computation of all these types of DFRFTs.

## 2 Mathematical Background

The normalized DFT matrix of size  $N$  is defined as follows:

$$\mathbf{F}_N = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & \dots & 1 & 1 \\ 1 & w_N^1 & \dots & w_N^{N-2} & w_N^{N-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & w_N^{N-2} & \dots & w_N^{(N-2)^2} & w_N^{(N-2)(N-1)} \\ 1 & w_N^{N-1} & \dots & w_N^{(N-1)(N-2)} & w_N^{(N-1)^2} \end{bmatrix} \tag{1}$$

where  $j$  is the imaginary unit,  $w_N = e^{-j \frac{2\pi}{N}}$ , and  $1/\sqrt{N}$  is the normalization scaling factor. Since  $\mathbf{F}_N$  is a symmetric matrix and  $\mathbf{F}_N \mathbf{F}_N^* = \mathbf{I}_N$ ,  $\mathbf{F}_N$  is a unitary matrix (symbol \* means Hermitian transpose and  $\mathbf{I}_N$  denotes an identity matrix). This implies the following properties [6]: (1) all the eigenvalues of  $\mathbf{F}_N$  are nonzero and have magnitude one, and (2) there exists a complete set of  $N$ -orthonormal eigenvectors, so we can write

$$\mathbf{F}_N = \mathbf{Z}_N \mathbf{\Lambda}_N \mathbf{Z}_N^T \tag{2}$$

where  $\mathbf{\Lambda}_N$  is a diagonal matrix of size  $N$ , whose diagonal entries are the eigenvalues of  $\mathbf{F}_N$  and  $\mathbf{Z}_N$  is the matrix whose columns are normalized mutually orthogonal eigenvectors of the matrix  $\mathbf{F}_N$ . The eigenvector  $\mathbf{z}_N^{(k)}$  corresponds to the eigenvalue  $\lambda_k$ . Since the matrix  $\mathbf{F}_N$  satisfies the property  $\mathbf{F}_N^4 = \mathbf{I}_N$ , each of its eigenvalues have to fulfil equation  $\lambda_k^4 = 1$ , so the DFT matrix has only four distinct eigenvalues:  $1, -1, j, -j$ . The multiplicities of these eigenvalues are well known [13], so for  $N \geq 4$  the eigenvalues are degenerated. This means that the set of eigenvectors is not unique. For this reason, it is necessary to specify a particular eigenvector set, which will be used in a definition of DFRFT.

The fractional power of matrix, including DFT matrix, can be obtained from its eigenvalue decomposition and the power of eigenvalues

$$\mathbf{F}_N^a = \mathbf{Z}_N \mathbf{\Lambda}_N^a \mathbf{Z}_N^T \tag{3}$$

where the fractional parameter  $a$  is real. Obviously, for  $a = 0$  the DFRFT matrix becomes the identity matrix, and for  $a = 1$  it is transformed into ordinary DFT

matrix. For completeness it should be added that an inverse discrete fractional Fourier transform (IDFRFT) matrix for a fractional parameter  $a$  is equal to the DFRFT matrix for a fractional parameter  $-a$ , so

$$(\mathbf{F}_N^a)^{-1} = \mathbf{F}_N^{-a} \tag{4}$$

The definition (3) of DFRFT was first introduced by Pei and Yeh [18, 19]. They defined the DFRFT in terms of a particular set of eigenvectors, which constitute the discrete counterpart of the set of Hermite–Gaussians functions (these functions are well-known eigenfunctions of FT and the fractional Fourier transform was defined through a spectral expansion in this base [14]). This idea was developed in work [3]. An important property of the eigenvectors of the DFT matrix is that they are either even or odd vectors [13]. Because of periodicity, the arguments are interpreted modulo  $N$  (as in the ordinary DFT), so a vector  $\mathbf{z}_N$  is even, when its coordinates (in standard basis) satisfy the equations

$$z_i = z_{N-i} \tag{5}$$

for  $i = 1, 2, \dots, \lfloor N/2 \rfloor$  ( $\lfloor x \rfloor$  is the greatest integer less than or equal to the argument  $x$ ). We are indexing the coordinates of the vector  $\mathbf{z}_N$  from 0 to  $N - 1$ . A vector  $\mathbf{z}_N$  is odd when its coordinates fulfil the following properties:

$$z_0 = 0 \quad \text{and} \quad z_i = -z_{N-i} \tag{6}$$

for  $i = 1, 2, \dots, \lfloor N/2 \rfloor$ .

### 3 Special Structure of DFRFT Matrix

In this paper we assume that the set of eigenvectors of the matrix  $\mathbf{F}_N$  has already been calculated, as it was shown in [3], and the eigenvectors are ordered according to the increasing number of zero-crossing. After normalization they form the matrix  $\mathbf{Z}_N$  which appears in Eqs. (2) and (3). We also assume that in the matrix  $\mathbf{\Lambda}_N$ , which occurs in these equations, the eigenvalues are arranged in the order of their associated eigenvectors. It should be noted that the DFRFT matrix calculated from (3) is symmetric because

$$f_{i,j}^{(a)} = \sum_{k=0}^{N-1} z_{i,k} \lambda_k^a z_{j,k} = \sum_{k=0}^{N-1} z_{j,k} \lambda_k^a z_{i,k} = f_{j,i}^{(a)} \tag{7}$$

for  $i, j = 0, 1, \dots, N - 1$ . Moreover, the DFRFT matrix has additional special properties, which result from the fact that each column of the matrix  $\mathbf{Z}_N$  is either an even or odd vector. One of them is that the first row (with index 0) is an even vector.

**Proposition 1** *The first row of the matrix  $\mathbf{F}_N^a$  is an even vector. It means that*

$$f_{0,j}^{(a)} = f_{0,N-j}^{(a)} \tag{8}$$

for  $j = 1, 2, \dots, \lfloor N/2 \rfloor$ .

*Proof*

$$f_{0,j}^{(a)} = \sum_{k=0}^{N-1} z_{0,k} \lambda_k^a z_{j,k} \tag{9}$$

$$f_{0,N-j}^{(a)} = \sum_{k=0}^{N-1} z_{0,k} \lambda_k^a z_{N-j,k} \tag{10}$$

If the column with index  $k$  of the matrix  $\mathbf{Z}_N$  is an even vector, then  $z_{j,k} = z_{N-j,k}$  for any  $j$ , hence,  $z_{0,k} \lambda_k^a z_{j,k} = z_{0,k} \lambda_k^a z_{N-j,k}$ . If the column with index  $k$  of the matrix  $\mathbf{Z}_N$  is an odd vector, then  $z_{0,k} = 0$  and also  $z_{0,k} \lambda_k^a z_{j,k} = 0 = z_{0,k} \lambda_k^a z_{N-j,k}$  for any  $j$ . Hence, the right side of Eq. (9) is equal to the right side of Eq. (10).  $\square$

Obviously, the first column of the matrix  $\mathbf{F}_N^a$  is an even vector too, because this matrix is symmetric.

**Proposition 2** *A matrix which we obtain after removing the first row and the first column from the matrix  $\mathbf{F}_N^a$  is persymmetric. It means that*

$$f_{i,j}^{(a)} = f_{N-j,N-i}^{(a)} \tag{11}$$

for  $i, j = 1, 2, \dots, N - 1$ .

*Proof*

$$f_{i,j}^{(a)} = \sum_{k=0}^{N-1} z_{i,k} \lambda_k^a z_{j,k} \tag{12}$$

$$f_{N-j,N-i}^{(a)} = \sum_{k=0}^{N-1} z_{N-j,k} \lambda_k^a z_{N-i,k} \tag{13}$$

If the column with index  $k$  of the matrix  $\mathbf{Z}_N$  is an even vector, then  $z_{j,k} = z_{N-j,k}$  for any  $j$ , hence,  $z_{i,k} \lambda_k^a z_{j,k} = z_{N-i,k} \lambda_k^a z_{N-j,k} = z_{N-j,k} \lambda_k^a z_{N-i,k}$  for any  $i$  and  $j$ . If the column with index  $k$  of the matrix  $\mathbf{Z}_N$  is an odd vector, then  $z_{j,k} = -z_{N-j,k}$  and also  $z_{i,k} \lambda_k^a z_{j,k} = -z_{N-i,k} \lambda_k^a (-z_{N-j,k}) = z_{N-j,k} \lambda_k^a z_{N-i,k}$ , hence, the right side of Eq. (12) is equal to the right side of Eq. (13).  $\square$

The two aforementioned properties of the matrix  $\mathbf{F}_N^a$  and its symmetry give it a special structure. We will write the matrix  $\mathbf{F}_N^a$  as a sum of three or two matrices and these matrices will have special structures as well. This trick may be useful to reduce the number of arithmetical operations when calculating a product of the matrix  $\mathbf{F}_N^a$  by a vector. The number of components of the sum depends on whether  $N$  is even or odd. If  $N$  is even, the matrix  $\mathbf{F}_N^a$  can be written as a sum of three matrices

$$\mathbf{F}_N^a = \mathbf{A}_N^{(a)} + \mathbf{B}_N^{(a)} + \mathbf{C}_N^{(a)} \tag{14}$$

where

$$\mathbf{A}_N^{(a)} = \begin{bmatrix} f_{0,0}^{(a)} & f_{0,1}^{(a)} & \cdots & f_{0,\frac{N}{2}-1}^{(a)} & f_{0,\frac{N}{2}}^{(a)} & f_{0,\frac{N}{2}-1}^{(a)} & \cdots & f_{0,1}^{(a)} \\ f_{0,1}^{(a)} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & & & & & & & \\ f_{0,\frac{N}{2}-1}^{(a)} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ f_{0,\frac{N}{2}}^{(a)} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ f_{0,\frac{N}{2}-1}^{(a)} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & & & & & & & \\ f_{0,1}^{(a)} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{15}$$

$$\mathbf{B}_N^{(a)} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & f_{1,1}^{(a)} & \cdots & f_{1,\frac{N}{2}-1}^{(a)} & 0 & f_{1,\frac{N}{2}+1}^{(a)} & \cdots & f_{1,N-1}^{(a)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & f_{1,\frac{N}{2}-1}^{(a)} & \cdots & f_{\frac{N}{2}-1,\frac{N}{2}-1}^{(a)} & 0 & f_{\frac{N}{2}-1,\frac{N}{2}+1}^{(a)} & \cdots & f_{1,\frac{N}{2}+1}^{(a)} \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & f_{1,\frac{N}{2}+1}^{(a)} & \cdots & f_{\frac{N}{2}-1,\frac{N}{2}+1}^{(a)} & 0 & f_{\frac{N}{2}-1,\frac{N}{2}-1}^{(a)} & \cdots & f_{1,\frac{N}{2}-1}^{(a)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & f_{1,N-1}^{(a)} & \cdots & f_{1,\frac{N}{2}+1}^{(a)} & 0 & f_{1,\frac{N}{2}-1}^{(a)} & \cdots & f_{1,1}^{(a)} \end{bmatrix} \tag{16}$$

$$\mathbf{C}_N^{(a)} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & f_{1,\frac{N}{2}}^{(a)} & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & \cdots & 0 & f_{\frac{N}{2}-1,\frac{N}{2}}^{(a)} & 0 & \cdots & 0 \\ 0 & f_{1,\frac{N}{2}}^{(a)} & \cdots & f_{\frac{N}{2}-1,\frac{N}{2}}^{(a)} & f_{\frac{N}{2},\frac{N}{2}}^{(a)} & f_{\frac{N}{2}-1,\frac{N}{2}}^{(a)} & \cdots & f_{1,\frac{N}{2}}^{(a)} \\ 0 & 0 & \cdots & 0 & f_{\frac{N}{2}-1,\frac{N}{2}}^{(a)} & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & \cdots & 0 & f_{1,\frac{N}{2}}^{(a)} & 0 & \cdots & 0 \end{bmatrix} \tag{17}$$

If  $N$  is odd, we can write the matrix  $\mathbf{F}_N^a$  as a sum of only two matrices

$$\mathbf{F}_N^a = \mathbf{A}_N^{(a)} + \mathbf{B}_N^{(a)} \tag{18}$$

where

$$\mathbf{A}_N^{(a)} = \begin{bmatrix} f_{0,0}^{(a)} & f_{0,1}^{(a)} & \cdots & f_{0,\frac{N-1}{2}}^{(a)} & f_{0,\frac{N-1}{2}}^{(a)} & \cdots & f_{0,1}^{(a)} \\ f_{0,1}^{(a)} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & & & & & & \\ f_{0,\frac{N-1}{2}}^{(a)} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ f_{0,\frac{N-1}{2}}^{(a)} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & & & & & & \\ f_{0,1}^{(a)} & 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{19}$$

$$\mathbf{B}_N^{(a)} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & f_{1,1}^{(a)} & \cdots & f_{1,\frac{N-1}{2}}^{(a)} & f_{1,\frac{N-1}{2}}^{(a)} & \cdots & f_{1,N-1}^{(a)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & f_{1,\frac{N-1}{2}}^{(a)} & \cdots & f_{\frac{N-1}{2},\frac{N-1}{2}}^{(a)} & f_{\frac{N-1}{2},\frac{N-1}{2}}^{(a)} & \cdots & f_{1,\frac{N-1}{2}}^{(a)} \\ 0 & f_{1,\frac{N-1}{2}}^{(a)} & \cdots & f_{\frac{N-1}{2},\frac{N-1}{2}}^{(a)} & f_{\frac{N-1}{2},\frac{N-1}{2}}^{(a)} & \cdots & f_{1,\frac{N-1}{2}}^{(a)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & f_{1,N-1}^{(a)} & \cdots & f_{1,\frac{N-1}{2}}^{(a)} & f_{1,\frac{N-1}{2}}^{(a)} & \cdots & f_{1,1}^{(a)} \end{bmatrix} \tag{20}$$

Example 1 presents the structures of DFRFT matrices and their components for a selected even number and a selected odd number  $N$ .

*Example 1* The structure of the matrix  $\mathbf{F}_N^a$  for  $N = 8$  is as follows:

$$\mathbf{F}_8^a = \begin{bmatrix} b & c & d & e & g & e & d & c \\ c & h & i & j & k & l & m & n \\ d & i & o & p & q & r & s & m \\ e & j & p & t & u & w & r & l \\ g & k & q & u & y & u & q & k \\ e & l & r & w & u & t & p & j \\ d & m & s & r & q & p & o & i \\ c & n & m & l & k & j & i & h \end{bmatrix} \tag{21}$$

where the entries:  $b, c, d, e, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, w, y$  are complex numbers, which are determined by  $N$  and the fractional parameter  $a$ . We can write the above matrix as a sum

$$\mathbf{F}_8^a = \mathbf{A}_8^{(a)} + \mathbf{B}_8^{(a)} + \mathbf{C}_8^{(a)} \tag{22}$$

where

$$\mathbf{A}_8^{(a)} = \begin{bmatrix} b & c & d & e & g & e & d & c \\ c & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ d & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ e & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ g & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ e & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ d & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ c & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (23)$$

$$\mathbf{B}_8^{(a)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & h & i & j & 0 & l & m & n \\ 0 & i & o & p & 0 & r & s & m \\ 0 & j & p & t & 0 & w & r & l \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & l & r & w & 0 & t & p & j \\ 0 & m & s & r & 0 & p & o & i \\ 0 & n & m & l & 0 & j & i & h \end{bmatrix} \quad (24)$$

$$\mathbf{C}_8^{(a)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & k & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & q & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & u & 0 & 0 & 0 \\ 0 & k & q & u & y & u & q & k \\ 0 & 0 & 0 & 0 & u & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & q & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & k & 0 & 0 & 0 \end{bmatrix} \quad (25)$$

The structure of the matrix  $\mathbf{F}_N^a$  for  $N = 7$  is as follows:

$$\mathbf{F}_7^a = \begin{bmatrix} b & c & d & e & e & d & c \\ c & g & h & i & j & k & l \\ d & h & m & n & o & p & k \\ e & i & n & q & r & o & j \\ e & j & o & r & q & n & i \\ d & k & p & o & n & m & h \\ c & l & k & j & i & h & g \end{bmatrix} \quad (26)$$

We can write this matrix as a sum

$$\mathbf{F}_7^a = \mathbf{A}_7^{(a)} + \mathbf{B}_7^{(a)} \quad (27)$$



where

$$\mathbf{A}_7^{(a)} = \begin{bmatrix} b & c & d & e & e & d & c \\ c & 0 & 0 & 0 & 0 & 0 & 0 \\ d & 0 & 0 & 0 & 0 & 0 & 0 \\ e & 0 & 0 & 0 & 0 & 0 & 0 \\ e & 0 & 0 & 0 & 0 & 0 & 0 \\ d & 0 & 0 & 0 & 0 & 0 & 0 \\ c & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (28)$$

$$\mathbf{B}_7^{(a)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & g & h & i & j & k & l \\ 0 & h & m & n & o & p & k \\ 0 & i & n & q & r & o & j \\ 0 & j & o & r & q & n & i \\ 0 & k & p & o & n & m & h \\ 0 & l & k & j & i & h & g \end{bmatrix} \quad (29)$$

#### 4 The Method of DFRFT Computing

Suppose you want to calculate the discrete fractional Fourier transform for the input vector  $\mathbf{x}_N$ . By  $\mathbf{y}_N^{(a)}$  we denote the output vector which is calculated using the formula

$$\mathbf{y}_N^{(a)} = \mathbf{F}_N^a \mathbf{x}_N \quad (30)$$

We assume that the matrix  $\mathbf{F}_N^a$  is given. To calculate the output vector  $\mathbf{y}_N^{(a)}$  it is necessary to perform  $N^2$  complex multiplications and  $N(N - 1)$  complex additions. However, if we use decomposition (14) when  $N$  is an even number or decomposition (18) when  $N$  is odd, the number of arithmetical operations required for calculating the discrete fractional Fourier transform can be significantly reduced. We can multiply each component of the sum by the input vector separately, and finally add the results. Let  $\mathbf{y}_N^{(A,a)}$  denote the product  $\mathbf{A}_N^{(a)} \mathbf{x}_N$ ,  $\mathbf{y}_N^{(B,a)}$ —the product  $\mathbf{B}_N^{(a)} \mathbf{x}_N$  and, if  $N$  is even,  $\mathbf{y}_N^{(C,a)}$ —the product  $\mathbf{C}_N^{(a)} \mathbf{x}_N$ . First, we will focus on calculating the product of the matrix  $\mathbf{A}_N^{(a)}$  by the input vector

$$\mathbf{y}_N^{(A,a)} = \mathbf{A}_N^{(a)} \mathbf{x}_N \quad (31)$$

If  $N$  is even, the matrix  $\mathbf{A}_N^{(a)}$  has the form (15) and

$$\mathbf{y}_N^{(A,a)} = \begin{bmatrix} f_{0,0}^{(a)}x_0 + f_{0,1}^{(a)}(x_1 + x_{N-1}) + \dots + f_{0,\frac{N}{2}-1}^{(a)}(x_{\frac{N}{2}-1} + x_{\frac{N}{2}+1}) + f_{0,\frac{N}{2}}^{(a)}x_{\frac{N}{2}} \\ f_{0,1}^{(a)}x_0 \\ \vdots \\ f_{0,\frac{N}{2}-1}^{(a)}x_0 \\ f_{0,\frac{N}{2}}^{(a)}x_0 \\ f_{0,\frac{N}{2}-1}^{(a)}x_0 \\ \vdots \\ f_{0,1}^{(a)}x_0 \end{bmatrix} \tag{32}$$

To make this calculation it is necessary to perform  $N - 1$  complex additions and  $N + 1$  complex multiplications.

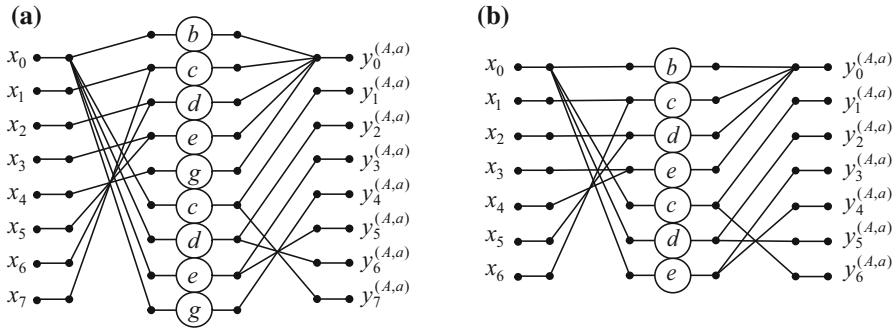
If  $N$  is odd, the matrix  $\mathbf{A}_N^{(a)}$  has the form (19) and

$$\mathbf{y}_N^{(A,a)} = \begin{bmatrix} f_{0,0}^{(a)}x_0 + f_{0,1}^{(a)}(x_1 + x_{N-1}) + \dots + f_{0,\frac{N-1}{2}}^{(a)}(x_{\frac{N-1}{2}} + x_{\frac{N+1}{2}}) \\ f_{0,1}^{(a)}x_0 \\ \vdots \\ f_{0,\frac{N-1}{2}}^{(a)}x_0 \\ f_{0,\frac{N-1}{2}}^{(a)}x_0 \\ \vdots \\ f_{0,1}^{(a)}x_0 \end{bmatrix} \tag{33}$$

To make this calculation it is necessary to perform  $N - 1$  complex additions and  $N$  complex multiplications. Example 2 presents the forms of vectors  $\mathbf{y}_N^{(A,a)}$  for  $N = 8$  and  $N = 7$ .

*Example 2* For  $N = 8$  the matrix  $\mathbf{A}_8^{(a)}$  is defined in (23). The product of this matrix by the input vector  $\mathbf{x}_8 = [x_0, x_1, \dots, x_7]^T$  will have the form

$$\mathbf{y}_8^{(A,a)} = \begin{bmatrix} bx_0 + c(x_1 + x_7) + d(x_2 + x_6) + e(x_3 + x_5) + gx_4 \\ cx_0 \\ dx_0 \\ ex_0 \\ gx_0 \\ ex_0 \\ dx_0 \\ cx_0 \end{bmatrix} \tag{34}$$



**Fig. 1** Data flow diagrams for calculation of vectors: **a**  $\mathbf{y}_8^{(A,a)}$  and **b**  $\mathbf{y}_7^{(A,a)}$

For  $N = 7$  the matrix  $\mathbf{A}_7^{(a)}$  is defined in (28). The product of this matrix by the input vector  $\mathbf{x}_7 = [x_0, x_1, \dots, x_6]^T$  will have the form

$$\mathbf{y}_7^{(A,a)} = \begin{bmatrix} bx_0 + c(x_1 + x_6) + d(x_2 + x_5) + e(x_3 + x_4) \\ cx_0 \\ dx_0 \\ ex_0 \\ ex_0 \\ dx_0 \\ cx_0 \end{bmatrix} \tag{35}$$

Figure 1 shows the data flow diagrams for the calculation of vectors  $\mathbf{y}_8^{(A,a)}$  and  $\mathbf{y}_7^{(A,a)}$ . In this paper, the data flow diagrams are oriented from left to right. Straight lines in the figures denote the operations of data transfer. We use lines without arrows not to clutter the picture. Points where lines converge denote summations. Note that the circles in this figure show the operations of multiplication by a complex numbers inscribed inside the circles.

Calculations of  $\mathbf{y}_N^{(A,a)}$  can be compactly described by appropriate matrix–vector procedures. If  $N$  is even, this procedure will be as follows:

$$\mathbf{y}_N^{(A,a)} = \mathbf{T}_{N \times (N+1)} \mathbf{V}_{N+1}^{(a)} \mathbf{X}_{(N+1) \times N} \mathbf{x}_N \tag{36}$$

where the matrix  $\mathbf{X}_{(N+1) \times N}$  is responsible for summing appropriate entries of the input vector:  $x_1 + x_{N-1}, x_2 + x_{N-2}, \dots, x_{N/2-1} + x_{N/2+1}$  and it has the form

$$\mathbf{X}_{(N+1) \times N} = \begin{bmatrix} 1 & \mathbf{0}_{1 \times (\frac{N}{2}-1)} & \mathbf{0} & \mathbf{0}_{1 \times (\frac{N}{2}-1)} \\ \mathbf{0}_{(\frac{N}{2}-1) \times 1} & \mathbf{1}_{\frac{N}{2}-1} & \mathbf{0}_{(\frac{N}{2}-1) \times 1} & \mathbf{J}_{\frac{N}{2}-1} \\ 0 & \mathbf{0}_{1 \times (\frac{N}{2}-1)} & \mathbf{1} & \mathbf{0}_{1 \times (\frac{N}{2}-1)} \\ \mathbf{1}_{\frac{N}{2} \times 1} & \mathbf{0}_{\frac{N}{2} \times (\frac{N}{2}-1)} & \mathbf{0}_{\frac{N}{2} \times 1} & \mathbf{0}_{\frac{N}{2} \times (\frac{N}{2}-1)} \end{bmatrix} \tag{37}$$

where the matrices  $\mathbf{0}_{m \times n}$  and  $\mathbf{1}_{m \times n}$  are matrices of size  $m \times n$  in which all the entries are equal to 0 or 1, respectively.  $\mathbf{J}_k$  is the exchange matrix of size  $k$  in which all the entries are zero, except those on the counter-diagonal going from the upper right corner to the lower left corner and all the counter-diagonal entries are equal to 1, i.e.

$$\mathbf{J}_k = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 0 & \dots & 0 & 0 \end{bmatrix} \tag{38}$$

The matrix  $\mathbf{V}_{N+1}^{(a)}$  which occurs in Eq. (36) is a diagonal matrix, which has the following form:

$$\mathbf{V}_{N+1}^{(a)} = \text{diag} \left( f_{0,0}^{(a)}, f_{0,1}^{(a)}, \dots, f_{0,\frac{N}{2}}^{(a)}, f_{0,1}^{(a)}, \dots, f_{0,\frac{N}{2}}^{(a)} \right) \tag{39}$$

The last matrix  $\mathbf{T}_{N \times (N+1)}$  which occurs in Eq. (36) has the following form:

$$\mathbf{T}_{N \times (N+1)} = \begin{bmatrix} \mathbf{1}_{1 \times (\frac{N}{2}+1)} & \mathbf{0}_{1 \times (\frac{N}{2}-1)} & \mathbf{0} \\ \mathbf{0}_{(\frac{N}{2}-1) \times (\frac{N}{2}+1)} & \mathbf{I}_{\frac{N}{2}-1} & \mathbf{0}_{(\frac{N}{2}-1) \times 1} \\ \mathbf{0}_{1 \times (\frac{N}{2}+1)} & \mathbf{0}_{1 \times (\frac{N}{2}-1)} & \mathbf{1} \\ \mathbf{0}_{(\frac{N}{2}-1) \times (\frac{N}{2}+1)} & \mathbf{J}_{\frac{N}{2}-1} & \mathbf{0}_{(\frac{N}{2}-1) \times 1} \end{bmatrix} \tag{40}$$

If  $N$  is odd, the matrix–vector procedure for calculating  $\mathbf{y}_N^{(A,a)}$  will have a bit different form

$$\mathbf{y}_N^{(A,a)} = \mathbf{T}_N \mathbf{V}_N^{(a)} \mathbf{X}_N \mathbf{x}_N \tag{41}$$

where the matrices which occur in (41) are as follows:

$$\mathbf{X}_N = \begin{bmatrix} \mathbf{1} & \mathbf{0}_{1 \times \frac{N-1}{2}} & \mathbf{0}_{1 \times \frac{N-1}{2}} \\ \mathbf{0}_{\frac{N-1}{2} \times 1} & \mathbf{I}_{\frac{N-1}{2}} & \mathbf{J}_{\frac{N-1}{2}} \\ \mathbf{1}_{\frac{N-1}{2} \times 1} & \mathbf{0}_{\frac{N-1}{2}} & \mathbf{0}_{\frac{N-1}{2}} \end{bmatrix} \tag{42}$$

$$\mathbf{V}_N^{(a)} = \text{diag} \left( f_{0,0}^{(a)}, f_{0,1}^{(a)}, \dots, f_{0,\frac{N-1}{2}}^{(a)}, f_{0,1}^{(a)}, \dots, f_{0,\frac{N-1}{2}}^{(a)} \right) \tag{43}$$

$$\mathbf{T}_N = \begin{bmatrix} \mathbf{1}_{1 \times \frac{N+1}{2}} & \mathbf{0}_{1 \times \frac{N-1}{2}} \\ \mathbf{0}_{\frac{N-1}{2} \times \frac{N+1}{2}} & \mathbf{I}_{\frac{N-1}{2}} \\ \mathbf{0}_{\frac{N-1}{2} \times \frac{N+1}{2}} & \mathbf{J}_{\frac{N-1}{2}} \end{bmatrix} \tag{44}$$

The matrix–vector procedures for calculation  $\mathbf{y}_8^{(A,a)}$  and  $\mathbf{y}_7^{(A,a)}$  are presented in Example 3. This example shows explicate forms of matrices which occur in these procedures, assuming that the matrices  $\mathbf{A}_8^{(a)}$  and  $\mathbf{A}_7^{(a)}$  are defined in (23) and (28), respectively.

*Example 3* For  $N = 8$  the matrix–vector procedure (36) for calculation  $\mathbf{y}_8^{(A,a)}$  will have the form

$$\mathbf{y}_8^{(A,a)} = \mathbf{T}_{8 \times 9} \mathbf{V}_9^{(a)} \mathbf{X}_{9 \times 8} \mathbf{x}_8 \tag{45}$$

where the matrices are as follows:

$$\mathbf{X}_{9 \times 8} = \begin{bmatrix} 1 & \mathbf{0}_{1 \times 3} & \mathbf{0} & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 1} & \mathbf{I}_3 & \mathbf{0}_{3 \times 1} & \mathbf{J}_3 \\ 0 & \mathbf{0}_{1 \times 3} & 1 & \mathbf{0}_{1 \times 3} \\ \mathbf{1}_{4 \times 1} & \mathbf{0}_{4 \times 3} & \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 3} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{46}$$

$$\mathbf{V}_9^{(a)} = \text{diag}(b, c, d, e, g, c, d, e, g) \tag{47}$$

$$\mathbf{T}_{8 \times 9} = \begin{bmatrix} \mathbf{1}_{1 \times 5} & \mathbf{0}_{1 \times 3} & \mathbf{0} \\ \mathbf{0}_{3 \times 5} & \mathbf{I}_3 & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{1 \times 5} & \mathbf{0}_{1 \times 3} & 1 \\ \mathbf{0}_{3 \times 5} & \mathbf{J}_3 & \mathbf{0}_{3 \times 1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{48}$$

For  $N = 7$  the matrix–vector procedure (41) for calculation  $\mathbf{y}_7^{(A,a)}$  will have the form

$$\mathbf{y}_7^{(A,a)} = \mathbf{T}_7 \mathbf{V}_7^{(a)} \mathbf{X}_7 \mathbf{x}_7 \tag{49}$$

where the matrices are as follows:

$$\mathbf{X}_7 = \begin{bmatrix} 1 & \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 1} & \mathbf{I}_3 & \mathbf{J}_3 \\ \mathbf{1}_{3 \times 1} & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{50}$$

$$\mathbf{V}_7^{(a)} = \text{diag}(b, c, d, e, c, d, e) \tag{51}$$

$$\mathbf{T}_7 = \begin{bmatrix} \mathbf{1}_{1 \times 4} & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 4} & \mathbf{I}_3 \\ \mathbf{0}_{3 \times 4} & \mathbf{J}_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{52}$$

Now we will focus on the product of the matrix  $\mathbf{B}_N^{(a)}$  by the input vector

$$\mathbf{y}_N^{(B,a)} = \mathbf{B}_N^{(a)} \mathbf{x}_N \tag{53}$$

If  $N$  is even, the matrix  $\mathbf{B}_N^{(a)}$  has the form (16). We can see that  $y_0^{(B,a)} = y_{N/2}^{(B,a)} = 0$  and also coordinates  $x_0$  and  $x_{N/2}$  are not involved in this calculation. Because of the special structure of the matrix  $\mathbf{B}_N^{(a)}$ , the rest of coordinates of the vector  $\mathbf{y}_N^{(B,a)}$  are convenient to calculate pairwise:  $y_1^{(B,a)}$  with  $y_{N-1}^{(B,a)}$ ,  $y_2^{(B,a)}$  with  $y_{N-2}^{(B,a)}, \dots, y_{N/2-1}^{(B,a)}$  with  $y_{N/2+1}^{(B,a)}$ , because for  $k = 1, 2, \dots, N/2 - 1$  we can write

$$\begin{aligned} \begin{bmatrix} y_k^{(B,a)} \\ y_{N-k}^{(B,a)} \end{bmatrix} &= \begin{bmatrix} f_{1,k}^{(a)} & f_{1,N-k}^{(a)} \\ f_{1,N-k}^{(a)} & f_{1,k}^{(a)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_{N-1} \end{bmatrix} \\ &+ \begin{bmatrix} f_{2,k}^{(a)} & f_{2,N-k}^{(a)} \\ f_{2,N-k}^{(a)} & f_{2,k}^{(a)} \end{bmatrix} \begin{bmatrix} x_2 \\ x_{N-2} \end{bmatrix} + \dots \\ &+ \begin{bmatrix} f_{k, \frac{N}{2}-1}^{(a)} & f_{k, \frac{N}{2}+1}^{(a)} \\ f_{k, \frac{N}{2}+1}^{(a)} & f_{k, \frac{N}{2}-1}^{(a)} \end{bmatrix} \begin{bmatrix} x_{\frac{N}{2}-1} \\ x_{\frac{N}{2}+1} \end{bmatrix} \end{aligned} \tag{54}$$

All the square matrices in the above equation have a structure of type

$$\begin{bmatrix} v & z \\ z & v \end{bmatrix} \tag{55}$$

and such a matrix can be written as a product [1]:

$$\begin{bmatrix} v & z \\ z & v \end{bmatrix} = \frac{1}{2} \mathbf{H}_2 \begin{bmatrix} v+z & 0 \\ 0 & v-z \end{bmatrix} \mathbf{H}_2 \tag{56}$$

where

$$\mathbf{H}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

is the Hadamard matrix of size 2. If we take into account the above-mentioned relationship, then Eq. (54) will take the following form:

$$\begin{aligned} \begin{bmatrix} y_k^{(B,a)} \\ y_{N-k}^{(B,a)} \end{bmatrix} &= \frac{1}{2} \mathbf{H}_2 \left( \begin{bmatrix} f_{1,k}^{(a)} + f_{1,N-k}^{(a)} & 0 \\ 0 & f_{1,k}^{(a)} - f_{1,N-k}^{(a)} \end{bmatrix} \mathbf{H}_2 \begin{bmatrix} x_1 \\ x_{N-1} \end{bmatrix} \right) + \\ &+ \begin{bmatrix} f_{2,k}^{(a)} + f_{2,N-k}^{(a)} & 0 \\ 0 & f_{2,k}^{(a)} - f_{2,N-k}^{(a)} \end{bmatrix} \mathbf{H}_2 \begin{bmatrix} x_2 \\ x_{N-2} \end{bmatrix} + \dots \\ &+ \begin{bmatrix} f_{k, \frac{N}{2}-1}^{(a)} + f_{k, \frac{N}{2}+1}^{(a)} & 0 \\ 0 & f_{k, \frac{N}{2}-1}^{(a)} - f_{k, \frac{N}{2}+1}^{(a)} \end{bmatrix} \mathbf{H}_2 \begin{bmatrix} x_{\frac{N}{2}-1} \\ x_{\frac{N}{2}+1} \end{bmatrix} \end{aligned} \quad (57)$$

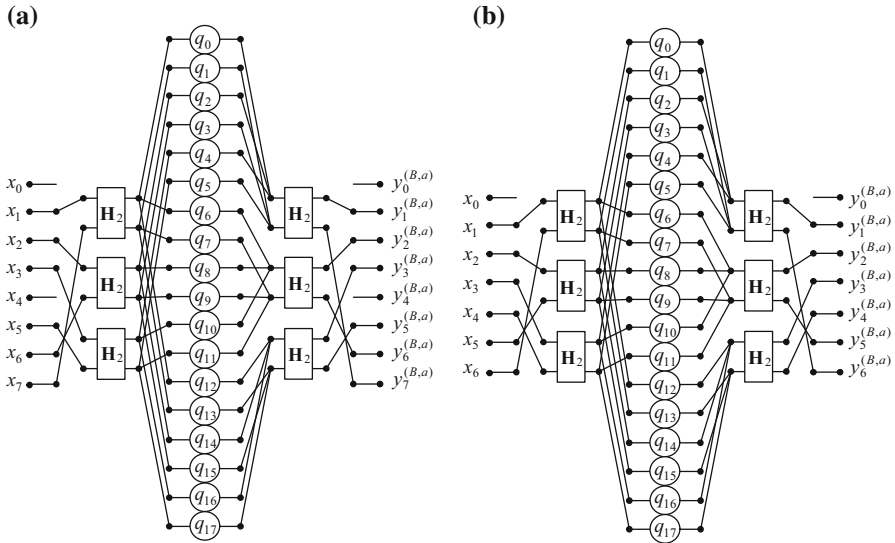
We should note that the last multiplication of the matrix  $\mathbf{H}_2$  by the sum of earlier calculated vectors (in parentheses) is executed only once. We should also note that when we calculate each subvector of the output vector first we have to multiply the Hadamard matrix  $\mathbf{H}_2$  by the subvectors:  $[x_1, x_{N-1}]^T, [x_2, x_{N-2}]^T, \dots, [x_{N/2-1}, x_{N/2+1}]^T$ , created from pairs of input coordinates. This allows us to perform these calculations only once and not to repeat them many times.

If  $N$  is odd, the matrix  $\mathbf{B}_N^{(a)}$  has the form (20). Looking at this matrix we can see that  $y_0^{(B,a)} = 0$  and also coordinate  $x_0$  is not involved in this calculation. The rest of coordinates of the vector  $\mathbf{y}_N^{(B,a)}$  are also convenient to calculate pairwise:  $y_1^{(B,a)}$  with  $y_{N-1}^{(B,a)}, y_2^{(B,a)}$  with  $y_{N-2}, \dots, y_{(N-1)/2}^{(B,a)}$  with  $y_{(N+1)/2}$ , because for  $k = 1, 2, \dots, (N-1)/2$  can write

$$\begin{aligned} \begin{bmatrix} y_k^{(B,a)} \\ y_{N-k}^{(B,a)} \end{bmatrix} &= \begin{bmatrix} f_{1,k}^{(a)} & f_{1,N-k}^{(a)} \\ f_{1,N-k}^{(a)} & f_{1,k}^{(a)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_{N-1} \end{bmatrix} \\ &+ \begin{bmatrix} f_{2,k}^{(a)} & f_{2,N-k}^{(a)} \\ f_{2,N-k}^{(a)} & f_{2,k}^{(a)} \end{bmatrix} \begin{bmatrix} x_2 \\ x_{N-2} \end{bmatrix} \dots \\ &+ \begin{bmatrix} f_{k, \frac{N-1}{2}}^{(a)} & f_{k, \frac{N+1}{2}}^{(a)} \\ f_{k, \frac{N+1}{2}}^{(a)} & f_{k, \frac{N-1}{2}}^{(a)} \end{bmatrix} \begin{bmatrix} x_{\frac{N-1}{2}} \\ x_{\frac{N+1}{2}} \end{bmatrix} \end{aligned} \quad (58)$$

All the square matrices in the equation above have the same structure as in (55), so we can write

$$\begin{aligned} \begin{bmatrix} y_k^{(B,a)} \\ y_{N-k}^{(B,a)} \end{bmatrix} &= \frac{1}{2} \mathbf{H}_2 \left( \begin{bmatrix} f_{1,k}^{(a)} + f_{1,N-k}^{(a)} & 0 \\ 0 & f_{1,k}^{(a)} - f_{1,N-k}^{(a)} \end{bmatrix} \mathbf{H}_2 \begin{bmatrix} x_1 \\ x_{N-1} \end{bmatrix} \right) \\ &+ \begin{bmatrix} f_{2,k}^{(a)} + f_{2,N-k}^{(a)} & 0 \\ 0 & f_{2,k}^{(a)} - f_{2,N-k}^{(a)} \end{bmatrix} \mathbf{H}_2 \begin{bmatrix} x_2 \\ x_{N-2} \end{bmatrix} + \dots \\ &+ \begin{bmatrix} f_{k, \frac{N-1}{2}}^{(a)} + f_{k, \frac{N+1}{2}}^{(a)} & 0 \\ 0 & f_{k, \frac{N-1}{2}}^{(a)} - f_{k, \frac{N+1}{2}}^{(a)} \end{bmatrix} \mathbf{H}_2 \begin{bmatrix} x_{\frac{N-1}{2}} \\ x_{\frac{N+1}{2}} \end{bmatrix} \end{aligned} \quad (59)$$



**Fig. 2** Data flow diagrams for calculation of vectors: **a**  $y_8^{(B,a)}$  and **b**  $y_7^{(B,a)}$

Using the approach presented above, we will show data flow diagrams for calculation of vectors  $y_8^{(B,a)}$  and  $y_7^{(B,a)}$ . For  $N = 8$  and  $N = 7$  the matrices  $\mathbf{B}_8^{(a)}$  and  $\mathbf{B}_7^{(a)}$  are defined in (24) and (29), respectively. Figure 2 shows the data flow diagrams for calculation of vectors  $y_8^{(B,a)}$  and  $y_7^{(B,a)}$ . Note that the rectangles in this figure show the operations of multiplication by the matrices inscribed inside the rectangles. In Fig. 2a the numbers  $q_i$  are equal to:  $q_0 = (h + n)/2, q_1 = (h - n)/2, q_2 = (i + m)/2, q_3 = (i - m)/2, q_4 = (j + l)/2, q_5 = (j - l)/2, q_6 = q_2, q_7 = q_3, q_8 = (o + s)/2, q_9 = (o - s)/2, q_{10} = (p + r)/2, q_{11} = (p - r)/2, q_{12} = q_4, q_{13} = q_5, q_{14} = q_{10}, q_{15} = q_{11}, q_{16} = (t + w)/2, q_{17} = (t - w)/2$ , where the numbers:  $h, n, \dots, w$  are the entries of the matrix  $\mathbf{B}_8^{(a)}$ . Similarly, in Fig. 2b the numbers  $q_i$  are equal to:  $q_0 = (g + l)/2, q_1 = (g - l)/2, q_2 = (h + k)/2, q_3 = (h - k)/2, q_4 = (i + j)/2, q_5 = (i - j)/2, q_6 = q_2, q_7 = q_3, q_8 = (m + p)/2, q_9 = (m - p)/2, q_{10} = (n + o)/2, q_{11} = (n - o)/2, q_{12} = q_4, q_{13} = q_5, q_{14} = q_{10}, q_{15} = q_{11}, q_{16} = (q + r)/2, q_{17} = (q - r)/2$ , where the numbers:  $g, l, \dots, r$  are the entries of the matrix  $\mathbf{B}_7^{(a)}$ .

Calculation of vector  $y_N^{(B,a)}$  can be compactly described by the suitable matrix–vector procedure. If  $N$  is even, this procedure will be as follows:

$$y_N^{(B,a)} = \mathbf{R}_{N \times (N-2)} \mathbf{W}_{(N-2) \times \frac{(N-2)^2}{2}} \mathbf{Q}_{\frac{(N-2)^2}{2}}^{(a)} \mathbf{U}_{\frac{(N-2)^2}{2} \times (N-2)} \mathbf{M}_{(N-2) \times N} \mathbf{x}_N \tag{60}$$

where the matrix  $\mathbf{M}_{(N-2) \times N}$ , which is responsible for reordering the coordinates of the input vector and multiplying the matrix  $\mathbf{H}_2$  by the subvectors  $[x_1, x_{N-1}]^T, [x_2, x_{N-2}]^T, \dots, [x_{N/2-1}, x_{N/2+1}]^T$ , has the form



$$\mathbf{M}_{(N-2) \times N} = \left[ \mathbf{0}_{(N-2) \times 1} \begin{matrix} \vdots \\ \mathbf{I}_{\frac{N-2}{2}} \\ \vdots \end{matrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{matrix} \vdots \\ \mathbf{0}_{(N-2) \times 1} \\ \vdots \end{matrix} \begin{matrix} \mathbf{J}_{\frac{N-2}{2}} \\ \otimes \\ \begin{bmatrix} 1 \\ -1 \end{bmatrix} \end{matrix} \right] \quad (61)$$

The symbol  $\otimes$  in the equation shown above denotes the Kronecker product operation. The matrix  $\mathbf{U}_{(N-2)^2/2 \times (N-2)}$  in Eq. (60), which is responsible for replication of the earlier calculated vector has the form

$$\mathbf{U}_{\frac{(N-2)^2}{2} \times (N-2)} = \mathbf{1}_{\frac{N-2}{2} \times 1} \otimes \mathbf{I}_{N-2} \quad (62)$$

The diagonal matrix  $\mathbf{Q}_{(N-2)^2/2}^{(a)}$  in Eq. (60) is responsible for multiplying the vector of the sums and differences of respective entries of the matrix (16) by the vector calculated earlier. The factor 1/2 is also included in this matrix. It has the form

$$\mathbf{Q}_{\frac{(N-2)^2}{2}}^{(a)} = \text{diag} \left( \frac{f_{1,1}^{(a)} + f_{1,N-1}^{(a)}}{2}, \frac{f_{1,1}^{(a)} - f_{1,N-1}^{(a)}}{2}, \right. \\ \frac{f_{1,2}^{(a)} + f_{1,N-2}^{(a)}}{2}, \frac{f_{1,2}^{(a)} - f_{1,N-2}^{(a)}}{2}, \\ \dots, \frac{f_{1,\frac{N}{2}-1}^{(a)} + f_{1,\frac{N}{2}+1}^{(a)}}{2}, \frac{f_{1,\frac{N}{2}-1}^{(a)} - f_{1,\frac{N}{2}+1}^{(a)}}{2}, \dots, \\ \left. \frac{f_{\frac{N}{2}-1,\frac{N}{2}-1}^{(a)} + f_{\frac{N}{2}-1,\frac{N}{2}+1}^{(a)}}{2}, \frac{f_{\frac{N}{2}-1,\frac{N}{2}-1}^{(a)} - f_{\frac{N}{2}-1,\frac{N}{2}+1}^{(a)}}{2} \right) \quad (63)$$

The matrix  $\mathbf{W}_{(N-2) \times (N-2)^2/2}$  in Eq. (60) is responsible for summation and has the form

$$\mathbf{W}_{(N-2) \times \frac{(N-2)^2}{2}} = \begin{bmatrix} \mathbf{I}_{\frac{N-2}{2}} \otimes \mathbf{1}_{1 \times \frac{N-2}{2}} \otimes [1 \ 0] \\ \mathbf{J}_{\frac{N-2}{2}} \otimes \mathbf{1}_{1 \times \frac{N-2}{2}} \otimes [0 \ 1] \end{bmatrix} \quad (64)$$

The last matrix  $\mathbf{R}_{N \times (N-2)}$  in Eq. (60) is responsible for multiplying the matrix  $\mathbf{H}_2$  by the subvectors of length 2 and reordering the coordinates. It has the form

$$\mathbf{R}_{N \times (N-2)} = \begin{bmatrix} \mathbf{0}_{1 \times (N-2)} & & & & \\ \mathbf{I}_{\frac{N-2}{2}} \otimes [1 \ 1] & & & & \\ & & & & \\ \mathbf{0}_{1 \times (N-2)} & & & & \\ \mathbf{J}_{\frac{N-2}{2}} \otimes [1 \ -1] & & & & \end{bmatrix} \quad (65)$$

To calculate the vector  $\mathbf{y}_N^{(B,a)}$  according to the procedure (60), it is necessary to perform  $N(N - 2)/2$  complex additions. From among these additions,  $N - 2$  are needed to multiply the matrix  $\mathbf{M}_{(N-2) \times N}$  by the input vector,  $(N/2 - 2)(N - 2)$  additions are needed to multiply the matrix  $\mathbf{W}_{(N-2) \times (N-2)^2/2}$  by earlier obtained vector, and  $N - 2$  are needed to multiply the matrix  $\mathbf{R}_{N \times (N-2)}$  by the last obtained vector. The additions

which were performed to obtain the matrix  $\mathbf{Q}_{(N-2)^2/2}^{(a)}$  were not counted, since this matrix may be prepared in advance. The calculation of the vector  $\mathbf{y}_N^{(B,a)}$  also requires  $(N - 2)^2/2$  complex multiplications. All these multiplications are needed to multiply the diagonal matrix  $\mathbf{Q}_{(N-2)^2/2}^{(a)}$  by the appropriate vector.

If  $N$  is odd, the procedure for calculation of vector  $\mathbf{y}_N^{(B,a)}$  will be as follows:

$$\mathbf{y}_N^{(B,a)} = \mathbf{R}_{N \times (N-1)} \mathbf{W}_{(N-1) \times \frac{(N-1)^2}{2}} \mathbf{Q}_{\frac{(N-1)^2}{2}}^{(a)} \mathbf{U}_{\frac{(N-1)^2}{2} \times (N-1)} \mathbf{M}_{(N-1) \times N} \mathbf{x}_N \quad (66)$$

where the matrices that occur in Eq. (66) have the forms

$$\mathbf{M}_{(N-1) \times N} = \left[ \mathbf{0}_{(N-1) \times 1} \mid \mathbf{I}_{\frac{N-1}{2}} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} \mid \mathbf{J}_{\frac{N-1}{2}} \otimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \right] \quad (67)$$

$$\mathbf{U}_{\frac{(N-1)^2}{2} \times (N-1)} = \mathbf{1}_{\frac{N-1}{2} \times 1} \otimes \mathbf{I}_{N-1} \quad (68)$$

$$\mathbf{Q}_{\frac{(N-1)^2}{2}}^{(a)} = \text{diag} \left( \frac{f_{1,1}^{(a)} + f_{1,N-1}^{(a)}}{2}, \frac{f_{1,1}^{(a)} - f_{1,N-1}^{(a)}}{2}, \frac{f_{1,2}^{(a)} + f_{1,N-2}^{(a)}}{2}, \frac{f_{1,2}^{(a)} - f_{1,N-2}^{(a)}}{2}, \dots, \frac{f_{1, \frac{N-1}{2}}^{(a)} + f_{1, \frac{N+1}{2}}^{(a)}}{2}, \frac{f_{1, \frac{N-1}{2}}^{(a)} - f_{1, \frac{N+1}{2}}^{(a)}}{2}, \dots, \frac{f_{\frac{N-1}{2}, \frac{N-1}{2}}^{(a)} + f_{\frac{N-1}{2}, \frac{N+1}{2}}^{(a)}}{2}, \frac{f_{\frac{N-1}{2}, \frac{N-1}{2}}^{(a)} - f_{\frac{N-1}{2}, \frac{N+1}{2}}^{(a)}}{2} \right) \quad (69)$$

$$\mathbf{W}_{(N-1) \times \frac{(N-1)^2}{2}} = \left[ \begin{array}{c} \mathbf{I}_{\frac{N-1}{2}} \otimes \mathbf{1}_{1 \times \frac{N-1}{2}} \otimes [1 \ 0] \\ \mathbf{J}_{\frac{N-1}{2}} \otimes \mathbf{1}_{1 \times \frac{N-1}{2}} \otimes [0 \ 1] \end{array} \right] \quad (70)$$

$$\mathbf{R}_{N \times (N-1)} = \left[ \begin{array}{c} \mathbf{0}_{1 \times (N-1)} \\ \mathbf{I}_{\frac{N-1}{2}} \otimes [1 \ 1] \\ \mathbf{J}_{\frac{N-1}{2}} \otimes [1 \ -1] \end{array} \right] \quad (71)$$

To calculate the vector  $\mathbf{y}_N^{(B,a)}$  according to the procedure (66), it is necessary to perform  $(N + 1)(N - 1)/2$  complex additions. From among these additions  $N - 1$  are needed to multiply the matrix  $\mathbf{M}_{(N-1) \times N}$  by the input vector,  $(N - 3)(N - 1)/2$  are needed to multiply the matrix  $\mathbf{W}_{(N-1) \times (N-1)^2/2}$  by earlier obtained vector, and  $N - 1$  are needed to multiply the matrix  $\mathbf{R}_{N \times (N-1)}$  by the last obtained vector. This procedure

also requires  $(N - 1)^2/2$  complex multiplications. All these multiplications are needed to multiply the diagonal matrix  $\mathbf{Q}_{(N-1)^2/2}^{(a)}$  by the appropriate vector.

The matrix–vector procedures for the calculation of  $\mathbf{y}_N^{(B,a)}$  for  $N = 8$  and  $N = 7$  are presented in Example 4. This example also demonstrates the explicate forms of matrices that occur in these procedures, assuming that the matrices  $\mathbf{B}_8^{(a)}$  and  $\mathbf{B}_7^{(a)}$  are defined in (24) and (29), respectively.

*Example 4* For  $N = 8$  the matrix–vector procedure (60) for calculating  $\mathbf{y}_8^{(B,a)}$  will have the form

$$\mathbf{y}_8^{(B,a)} = \mathbf{R}_{8 \times 6} \mathbf{W}_{6 \times 18} \mathbf{Q}_{18}^{(a)} \mathbf{U}_{18 \times 6} \mathbf{M}_{6 \times 8} \mathbf{x}_8 \tag{72}$$

where the matrices which occur in above presented equation are as follows:

$$\begin{aligned} \mathbf{M}_{6 \times 8} &= \left[ \mathbf{0}_{6 \times 1} \Big| \mathbf{I}_3 \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Big| \mathbf{0}_{6 \times 1} \Big| \mathbf{J}_3 \otimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \right] \\ &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \end{bmatrix} \end{aligned} \tag{73}$$

$$\mathbf{U}_{18 \times 6} = \mathbf{1}_{3 \times 1} \otimes \mathbf{I}_6 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{74}$$

$$\mathbf{Q}_{18}^{(a)} = \text{diag} \left( \frac{h+n}{2}, \frac{h-n}{2}, \frac{i+m}{2}, \frac{i-m}{2}, \dots \right)$$

$$\left( \frac{j+l}{2}, \frac{j-l}{2}, \frac{i+m}{2}, \frac{i-m}{2}, \frac{o+s}{2}, \frac{o-s}{2}, \frac{p+r}{2}, \frac{p-r}{2}, \frac{j+l}{2}, \frac{j-l}{2}, \frac{p+r}{2}, \frac{p-r}{2}, \frac{t+w}{2}, \frac{t-w}{2} \right) \tag{75}$$

$$\begin{aligned} \mathbf{W}_{6 \times 18} &= \begin{bmatrix} \mathbf{I}_3 \otimes \mathbf{1}_{1 \times 3} \otimes [1 \ 0] \\ \mathbf{J}_3 \otimes \mathbf{1}_{1 \times 3} \otimes [0 \ 1] \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{76}$$

$$\mathbf{R}_{8 \times 6} = \begin{bmatrix} \mathbf{0}_{1 \times 6} \\ \mathbf{I}_3 \otimes [1 \ -1] \\ \mathbf{0}_{1 \times 6} \\ \mathbf{J}_3 \otimes [1 \ -1] \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{77}$$

For  $N = 7$  the matrix–vector procedure (66) for calculating  $\mathbf{y}_7^{(B,a)}$  will have the form

$$\mathbf{y}_7^{(B,a)} = \mathbf{R}_{7 \times 6} \mathbf{W}_{6 \times 18} \mathbf{Q}_{18}^{(a)} \mathbf{U}_{18 \times 6} \mathbf{M}_{6 \times 7} \mathbf{x}_7 \tag{78}$$

where the matrices which occur in the equation presented above are as follows:

$$\begin{aligned} \mathbf{M}_{6 \times 7} &= \begin{bmatrix} \mathbf{0}_{6 \times 1} \otimes \mathbf{I}_3 \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ \mathbf{J}_3 \otimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \end{bmatrix} \\ &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix} \end{aligned} \tag{79}$$

The matrix  $\mathbf{U}_{18 \times 6}$  is the same as in Eq. (72).

$$\mathbf{Q}_{18}^{(a)} = \text{diag} \left( \frac{g+l}{2}, \frac{g-l}{2}, \frac{h+k}{2}, \frac{h-k}{2}, \frac{i+j}{2}, \frac{i-j}{2}, \frac{h+k}{2}, \frac{h-k}{2}, \frac{m+p}{2}, \frac{m-p}{2}, \frac{n+o}{2}, \frac{n-o}{2}, \frac{i+j}{2}, \frac{i-j}{2}, \frac{n+o}{2}, \frac{n-o}{2}, \frac{q+r}{2}, \frac{q-r}{2} \right) \tag{80}$$

The matrix  $\mathbf{W}_{6 \times 18}$  is the same as in Eq. (72).

$$\mathbf{R}_{7 \times 6} = \begin{bmatrix} \mathbf{0}_{1 \times 6} \\ \mathbf{I}_3 \otimes [\bar{1} \ \bar{1}] \\ \mathbf{J}_3 \otimes [\bar{1} \ -\bar{1}] \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{81}$$

The last component  $\mathbf{C}_N^{(a)}$  appears in decomposition (14) only if  $N$  is even. Now we will focus on the product of the matrix  $\mathbf{C}_N^{(a)}$  by the input vector

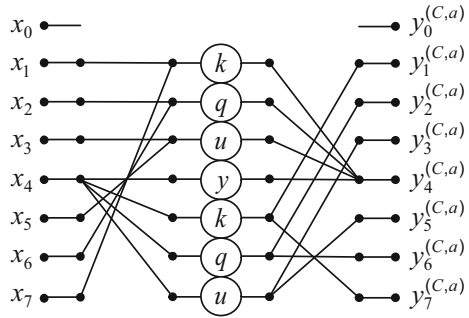
$$\mathbf{y}_N^{(C,a)} = \mathbf{C}_N^{(a)} \mathbf{x}_N \tag{82}$$

For even  $N$  the matrix  $\mathbf{C}_N^{(a)}$  has the form (17) and

$$\mathbf{y}_N^{(C,a)} = \begin{bmatrix} 0 \\ f_{1, \frac{N}{2}}^{(a)} x_{\frac{N}{2}} \\ f_{2, \frac{N}{2}}^{(a)} x_{\frac{N}{2}} \\ \vdots \\ f_{\frac{N}{2}-1, \frac{N}{2}}^{(a)} x_{\frac{N}{2}} \\ f_{1, \frac{N}{2}}^{(a)} (x_1 + x_{N-1}) + \dots + f_{\frac{N}{2}-1, \frac{N}{2}}^{(a)} (x_{\frac{N}{2}-1} + x_{\frac{N}{2}+1}) + f_{\frac{N}{2}, \frac{N}{2}}^{(a)} x_{\frac{N}{2}} \\ f_{\frac{N}{2}-1, \frac{N}{2}}^{(a)} x_{\frac{N}{2}} \\ \vdots \\ f_{1, \frac{N}{2}}^{(a)} x_{\frac{N}{2}} \end{bmatrix} \tag{83}$$

To make this calculation it is necessary to perform  $N - 2$  complex additions and  $N - 1$  complex multiplications. Example 5 presents the form of the vector  $\mathbf{y}_N^{(C,a)}$  for  $N = 8$ .

**Fig. 3** Data flow diagrams for calculation of vector  $\mathbf{y}_8^{(C,a)}$



*Example 5* For  $N = 8$  the matrix  $\mathbf{C}_8^{(a)}$  is defined in (25). The product of this matrix by the input vector  $\mathbf{x}_8 = [x_0, x_1, \dots, x_7]^T$  will have the form

$$\mathbf{y}_8^{(C,a)} = \begin{bmatrix} 0 \\ kx_4 \\ qx_4 \\ ux_4 \\ k(x_1 + x_7) + q(x_2 + x_6) + u(x_3 + x_5) + yx_4 \\ ux_4 \\ qx_4 \\ kx_4 \end{bmatrix} \tag{84}$$

Figure 3 shows the data flow diagram for calculation of vector  $\mathbf{y}_8^{(C,a)}$ .

Calculation  $\mathbf{y}_N^{(C,a)}$  can be concisely described by appropriate matrix–vector procedure. This procedure will be as follows:

$$\mathbf{y}_N^{(C,a)} = \mathbf{K}_{N \times (N-1)} \mathbf{G}_{N-1}^{(a)} \mathbf{L}_{(N-1) \times N} \mathbf{x}_N \tag{85}$$

where the matrix  $\mathbf{L}_{(N-1) \times N}$  is responsible for summing the appropriate entries of the input vector:  $x_1 + x_{N-1}, x_2 + x_{N-2}, \dots, x_{N/2-1} + x_{N/2+1}$  and it has the form

$$\mathbf{L}_{(N-1) \times N} = \begin{bmatrix} \mathbf{0}_{(\frac{N}{2}-1) \times 1} & \mathbf{I}_{\frac{N}{2}-1} & \mathbf{0}_{(\frac{N}{2}-1) \times 1} & \mathbf{J}_{\frac{N}{2}-1} \\ \mathbf{0}_{\frac{N}{2} \times 1} & \mathbf{0}_{\frac{N}{2} \times (\frac{N}{2}-1)} & \mathbf{1}_{\frac{N}{2} \times 1} & \mathbf{0}_{\frac{N}{2} \times (\frac{N}{2}-1)} \end{bmatrix} \tag{86}$$

The matrix  $\mathbf{G}_{N-1}^{(a)}$  that occurs in Eq. (85) is a diagonal matrix, which has the following form:

$$\mathbf{G}_{N-1}^{(a)} = \text{diag} \left( f_{1, \frac{N}{2}}^{(a)}, f_{2, \frac{N}{2}}^{(a)}, \dots, f_{\frac{N}{2}, \frac{N}{2}}^{(a)}, f_{1, \frac{N}{2}}^{(a)}, f_{2, \frac{N}{2}}^{(a)}, \dots, f_{\frac{N}{2}-1, \frac{N}{2}}^{(a)} \right) \tag{87}$$

The last matrix  $\mathbf{K}_{N \times (N-1)}$  which occurs in Eq. (85) has the form

$$\mathbf{K}_{N \times (N-1)} = \begin{bmatrix} \mathbf{0}_{1 \times \frac{N}{2}} & \mathbf{0}_{1 \times (\frac{N}{2}-1)} \\ \mathbf{0}_{(\frac{N}{2}-1) \times \frac{N}{2}} & \mathbf{I}_{\frac{N}{2}-1} \\ \mathbf{1}_{1 \times \frac{N}{2}} & \mathbf{0}_{1 \times (\frac{N}{2}-1)} \\ \mathbf{0}_{(\frac{N}{2}-1) \times \frac{N}{2}} & \mathbf{J}_{\frac{N}{2}-1} \end{bmatrix} \tag{88}$$

The matrix–vector procedure (85) for calculating  $\mathbf{y}_N^{(C,a)}$  for  $N = 8$  is presented in Example 6. This example also shows explicate forms of matrices which occur in this procedure, assuming that the matrix  $\mathbf{C}_8^{(a)}$  is defined in (25).

*Example 6* For  $N = 8$  the matrix–vector procedure (85) for calculating  $\mathbf{y}_8^{(C,a)}$  will have the form

$$\mathbf{y}_8^{(C,a)} = \mathbf{K}_{8 \times 7} \mathbf{G}_7^{(a)} \mathbf{L}_{7 \times 8} \mathbf{x}_8 \tag{89}$$

where the matrices are as follows:

$$\mathbf{L}_{7 \times 8} = \begin{bmatrix} \mathbf{0}_{3 \times 1} & \mathbf{I}_3 & \mathbf{0}_{3 \times 1} & \mathbf{J}_3 \\ \mathbf{0}_{4 \times 1} & \mathbf{0}_{4 \times 3} & \mathbf{1}_{4 \times 1} & \mathbf{0}_{4 \times 3} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{90}$$

$$\mathbf{G}_7^{(a)} = \text{diag}(k, q, u, y, k, q, u) \tag{91}$$

$$\mathbf{K}_{8 \times 7} = \begin{bmatrix} \mathbf{0}_{1 \times 4} & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 4} & \mathbf{I}_3 \\ \mathbf{1}_{1 \times 4} & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 4} & \mathbf{J}_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{92}$$

To obtain the final output vector  $\mathbf{y}_N^{(a)}$ , namely, the discrete fractional Fourier transform defined in (30), we have to add vectors  $\mathbf{y}_N^{(A,a)}$ ,  $\mathbf{y}_N^{(B,a)}$  and also  $\mathbf{y}_N^{(C,a)}$  if  $N$  is even. For even  $N$  we have  $y_0^{(B,a)} = y_0^{(C,a)} = 0$ , so to obtain  $y_0^{(a)}$  we do not need to

**Table 1** Number of additions and multiplications for even  $N$

	$\mathbf{y}_N^{(A,a)}$	$\mathbf{y}_N^{(B,a)}$	$\mathbf{y}_N^{(C,a)}$	Summing	Total
Additions	$N - 1$	$\frac{N(N-2)}{2}$	$N - 2$	$2N - 3$	$\frac{N^2}{2} + 3N - 6$
Multiplications	$N + 1$	$\frac{(N-2)^2}{2}$	$N - 1$	0	$\frac{N^2}{2} + 2$

**Table 2** Number of additions and multiplications for odd  $N$

	$\mathbf{y}_N^{(A,a)}$	$\mathbf{y}_N^{(B,a)}$	Summing	Total
Additions	$N - 1$	$\frac{(N-1)(N+1)}{2}$	$N - 1$	$\frac{N^2-1}{2} + 2N - 2$
Multiplications	$N$	$\frac{(N-1)^2}{2}$	0	$\frac{N^2+1}{2}$

perform any additions. Also, since  $y_{N/2}^{(B,a)} = 0$ , it is necessary to perform only one addition to obtain  $y_{N/2}^{(a)}$ . For other coordinates  $y_i^{(a)}$ , where  $i \neq 0$  and  $i \neq N/2$ , we have to perform two additions. The whole number of additions when summing vectors  $\mathbf{y}_N^{(A,a)}$ ,  $\mathbf{y}_N^{(B,a)}$  and  $\mathbf{y}_N^{(C,a)}$  is equal to  $2(N - 2) + 1 = 2N - 3$ . If  $N$  is odd we have to add only two vectors:  $\mathbf{y}_N^{(A,a)}$  and  $\mathbf{y}_N^{(B,a)}$ . Since  $y_0^{(B,a)} = 0$ , we do not need to perform any additions to obtain  $y_0^{(a)}$ . For other coordinates  $y_i^{(a)}$ , where  $i \neq 0$  we have to perform one addition. The whole number of additions when summing vectors  $\mathbf{y}_N^{(A,a)}$  and  $\mathbf{y}_N^{(B,a)}$  is equal to  $N - 1$ .

### 5 Computational Complexity

Let us assess the computational complexity in terms of the number of multiplications and additions required to obtain DFRFT. Direct calculation of the discrete fractional Fourier transform for an input vector  $\mathbf{x}_N$ , assuming that the matrix  $\mathbf{F}_N^a$  defined by (3) is given, requires  $N^2$  multiplications of a complex number and  $N(N - 1)$  complex additions.

If we use the decomposition of the matrix  $\mathbf{F}_N^a$  into three or two matrices, according to (14) or (18), respectively, then the vectors  $\mathbf{y}_N^{(A,a)}$ ,  $\mathbf{y}_N^{(B,a)}$  and  $\mathbf{y}_N^{(C,a)}$  if  $N$  is even are calculated. When, in the end, they are added, the total number of additions and multiplications will be smaller. We will evaluate the number of additions and multiplications by calculating vectors  $\mathbf{y}_N^{(A,a)}$ ,  $\mathbf{y}_N^{(B,a)}$  and  $\mathbf{y}_N^{(C,a)}$  if  $N$  is even, using our method. These vectors and the total number of the operations, which are needed to obtain the resulting vector  $\mathbf{y}_N^{(a)}$ , will be summed. These calculations will be performed for an even and odd  $N$  separately and the results will be presented in Tables 1 and 2, respectively.

Table 3 presents the number of additions and multiplications, which are necessary for calculating DFRFT transform for the input vector of even length  $N$ , using the direct method, the method from [22] and the method proposed. Since the method from work



**Table 3** Comparison of number of arithmetical operations for even length DFRFT

$N$	Additions			Multiplications		
	Direct method	Method from [22]	Proposed method	Direct method	Method from [22]	Proposed method
2	2	4	2	4	6	4
4	22	10	14	16	14	10
6	30	20	30	36	26	20
8	56	34	50	64	42	34
10	90	52	74	100	62	52
12	132	74	102	144	86	74
14	182	100	134	196	114	100
16	240	130	170	256	146	130
18	306	164	210	324	182	164
20	380	202	254	400	222	202

**Table 4** Comparison of number of arithmetical operations for odd length DFRFT

$N$	Additions		Multiplications	
	Direct method	Proposed method	Direct method	Proposed method
3	6	8	9	5
5	20	20	25	13
7	42	36	49	25
9	72	56	81	41
11	110	80	121	61
13	156	108	169	85
15	210	140	225	113
17	272	176	289	145
19	342	216	361	181
21	420	260	441	221

[22] cannot be used for the input vector of odd length, Table 4 presents the number of additions and multiplications, which are necessary for calculating DFRFT transform for the input vector of odd length  $N$ , using only the direct method and the method proposed.

After the analysis of Tables 3 and 4, it can be seen that the number of multiplications in the method proposed is almost twice as small as that in the direct method of calculating DFRFT. This observation is true for vectors of both even and odd lengths. The number of additions in our method is also smaller than in the direct method. This difference in favour of our method increases with the length of the input vector.

For an input signal of an even length we can compare our method with the method from [22]. As it can be seen in Table 3, the number of additions is lower in the method from [22], but the number of multiplications is lower in the method proposed.

## 6 Conclusions

In this paper, we proposed a new approach to the design of the reduced complexity algorithms for the discrete fractional Fourier transform computation. In the approach proposed, the original DFFT matrix can be decomposed as an algebraic sum of a dense matrix and of one or two another matrices which have many zero entries. Thus, the decomposition of the original matrix into submatrices can be represented as a sum of the partial products of each submatrix by the same input vector. The dense matrix possesses a unique structure that allows us to determine an effective factorization of this matrix and leads to accelerate computation by reducing the arithmetical complexity of a matrix–vector product. The calculation of the remaining matrix–vector products requires only a small number of arithmetic operations. Based on the matrix factorization and the Kronecker product, the effective method for the DFRFT computation has been derived. For the sake of simplicity, the two examples, for  $N = 7$  and  $N = 8$ , have been considered. However, it is clear that the approach proposed is applicable for any arbitrary case (regardless of whether the length of the input vector is odd or even).

**Acknowledgements** Funding was provided by West Pomeranian University of Technology Szczecin (PL).

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

1. A. Cariow, Strategies for the synthesis of fast algorithms for the computation of the matrix–vector products. *J. Signal Process. Theory Appl.* **3**(1), 1–19 (2014)
2. A. Cariow, D. Majorkowska-Mech, Fast algorithm for discrete fractional Hadamard transform. *Numer. Algorithms* **68**(3), 585–600 (2015)
3. Ç.C. Candan, M.A. Kutay, H.M. Ozaktas, The discrete fractional Fourier transform. *IEEE Trans. Signal Process.* **48**(5), 1329–1337 (2000)
4. B.W. Dickinson, K. Steiglitz, Eigenvectors and functions of the discrete Fourier transform. *IEEE Trans. Acoust. Speech Signal Process.* **30**(1), 25–31 (1982)
5. I. Djurović, S. Stanković, I. Pitas, Digital watermarking in the fractional Fourier transformation domain. *J. Netw. Comput. Appl.* **24**(2), 167–173 (2001)
6. P.R. Halmos, *Finite Dimensional Vector Spaces* (Princeton University Press, Princeton, 1947)
7. B. Hennelly, J.T. Sheridan, Fractional Fourier transform-based image encryption: phase retrieval algorithm. *Opt. Commun.* **226**(1–6), 61–80 (2003)
8. M. Irfan, L. Zheng, H. Shahzad, Review of computing algorithms for discrete fractional Fourier transform. *Res. J. Appl. Sci. Eng. Technol.* **6**(11), 1911–1919 (2013)
9. N. Jindal, K. Singh, Image and video processing using discrete fractional transforms. *SIViP* **8**(8), 1543–1553 (2014)
10. S. Liu, Q. Mi, B. Zhu, Optical image encryption with multistage and multichannel fractional Fourier-domain filtering. *Opt. Lett.* **26**(16), 1242–1244 (2001)

11. S. Liu, T. Shan, R. Tao, Y.-D. Zhang, G. Zhang, F. Zhang, Y. Wang, Sparse discrete fractional Fourier transform and its applications. *IEEE Trans. Signal Process.* **62**(24), 6582–6595 (2014)
12. Z. Liu, H. Zhao, S. Liu, A discrete fractional random transform. *Opt. Commun.* **255**(4–6), 357–365 (2005)
13. J.H. McClellan, T.W. Parks, Eigenvalue and eigenvector decomposition of the discrete Fourier transform. *IEEE Trans. Audio Electroacoust.* **20**(1), 66–74 (1972)
14. V. Namias, The fractional order Fourier transform and its application to quantum mechanics. *J. Inst. Appl. Math.* **25**, 241–265 (1980)
15. N.K. Nishchal, J. Joseph, K. Singh, Fully phase encryption using fractional Fourier transform. *Opt. Eng.* **42**(6), 1583–1588 (2003)
16. H.M. Ozaktas, O. Ankan, M.A. Kutay, G. Bozdagi, Digital computation of the fractional Fourier transform. *IEEE Trans. Signal Process.* **44**(9), 2141–2150 (1996)
17. S.-C. Pei, J.-J. Ding, Closed-form discrete fractional and affine Fourier transforms. *IEEE Trans. Signal Process.* **48**(5), 1338–1353 (2000)
18. S.-C. Pei, M.-H. Yeh, Discrete fractional Fourier transform. In: *Proceedings of IEEE International Symposium on Circuits and Systems* (1996), pp. 536–539
19. S.-C. Pei, M.-H. Yeh, Improved discrete fractional Fourier transform. *Opt. Lett.* **22**(14), 1047–1049 (1997)
20. S.-C. Pei, M.-H. Yeh, Discrete fractional Hadamard transform. In: *Proceedings of IEEE International Symposium on Circuits and Systems*, vol. 3 (1999), pp. 179–182
21. S.-C. Pei, M.-H. Yeh, Discrete fractional Hilbert transform. *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.* **47**(11), 1307–1311 (2000)
22. S.-C. Pei, M.H. Yeh, The discrete fractional cosine and sine transforms. *IEEE Trans. Signal Process.* **49**(6), 1198–1207 (2001)
23. S.-C. Pei, C.-C. Tseng, M.-H. Yeh, J.-J. Shyu, Discrete fractional Hartley and Fourier transforms. *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.* **45**(6), 665–675 (1998)
24. B. Santhanam, J.H. McClellan, Discrete rotational Fourier transform. *IEEE Trans. Signal Process.* **44**(4), 994–998 (1996)
25. R. Tao, G. Liang, X. Zhao, An efficient FPGA-based implementation of fractional Fourier transform algorithm. *J. Signal Process. Syst.* **60**(1), 47–58 (2010)
26. N. Wiener, Hermitian polynomials and Fourier analysis. *J. Math. Phys.* **8**, 70–73 (1929)
27. I.Ş. Yetik, M.A. Kutay, H.M. Ozaktas, Image representation and compression with the fractional Fourier transform. *Opt. Commun.* **197**(4–6), 275–278 (2001)