Circuits Syst Signal Process (2017) 36:2893–2916
DOI 10.1007/s00034-016-0447-8

Fractional Fourier, Hartley, Cosine and Sine Number-Theoretic Transforms Based on Matrix Functions

Paulo Hugo E. S. Lima[1] · **Juliano B. Lima**[1] ·
Ricardo M. Campello de Souza[1]

Received: 21 October 2015 / Revised: 20 October 2016 / Accepted: 22 October 2016 /
Published online: 1 November 2016

Abstract In this paper, we introduce fractional number-theoretic transforms (FrNTT) based on matrix functions. In contrast to previously proposed FrNTT, our approach does not require the construction of any number-theoretic transform (NTT) eigenvectors set. This allows us to obtain an FrNTT matrix by means of a closed-form expression corresponding to a linear combination of integer powers of the respective NTT matrix. Fractional Fourier, Hartley, cosine and sine number-theoretic transforms are developed. We show that fast algorithms applicable to ordinary NTT can also be used to compute the proposed FrNTT. Furthermore, we investigate the relationship between fractional Fourier and Hartley number-theoretic transforms, and demonstrate the applicability of the proposed FrNTT to a recently introduced image encryption scheme.

Keywords Number-theoretic transforms · Fractional transforms · Matrix functions · Image encryption

1 Introduction

The inaugural investigations related to fractional transforms date back to the first half of the twentieth century [11,39]. In such studies, the computation of a Fourier transform

✉ Juliano B. Lima
juliano_bandeira@ieee.org

Paulo Hugo E. S. Lima
paulohugos@gmail.com

Ricardo M. Campello de Souza
ricardo@ufpe.br

[1] Department of Electronics and Systems, Federal University of Pernambuco, Recife, Brazil

is viewed as the application of a linear operator, the *Fourier operator*, to a function (or signal) whose spectrum one desires to obtain; the fractional Fourier transform (FrFT) corresponds to a generalization, where noninteger powers of the Fourier operator can be considered. This enables a diversity of possibilities, which remained hidden until the 1980s, when the FrFT was rediscovered. In quantum mechanics, optics and signal processing, the FrFT has reappeared as a useful mathematical tool, with applications based on new theoretic results and interesting interpretations [2,27–29].

In the time–frequency plane, the computation of the FrFT of a signal in the time domain (horizontal axis to the right) can be viewed as a counterclockwise rotation by an angle $\alpha = a\pi/2, a \in \mathbb{R}$. If a is noninteger, the signal is taken to an intermediate domain, which represents, in the physical point of view, something like a *hybrid domain* between time and frequency [2]. In optics, the FrFT is related to systems whose output corresponds to the Fourier transform of the input signal [27,29]. These systems produce sequences of images of an object. In certain distances, the object itself as well as its Fourier transform is observed; in arbitrary distances, the FrFT of the object is obtained.

The fractional Fourier transform inspired the definition of fractional Hartley, cosine and sine transforms, among others [3,30]. Furthermore, discrete fractional transforms have also been introduced [8,31,34]. In particular, there are several approaches for constructing a discrete fractional Fourier transform (DFrFT) [8,32,37]. Besides allowing the employment of fast algorithms in its computation, a DFrFT should numerically approximate the corresponding continuous-time transform and preserve some of its properties.

More recently, fractional number-theoretic transforms (NTT) were proposed. In [33], for example, closed-form orthogonal NTT eigenvector sets are constructed from complete generalized Legendre sequences. Such sets are then used to spectrally expand the NTT matrix and to compute its fractional powers. The method introduced in [21] is also based on the spectral expansion of the NTT matrix. However, in this case, orthogonal NTT eigenvectors are obtained from an extension to the finite field scenario of a matrix which commutes with the NTT matrix; the eigenvectors derived through this approach can be viewed as finite field Hermite–Gaussian vectors.

In this paper, we generalize the ideas introduced in [25] and describe several types of new fractional number-theoretic transforms based on matrix functions [14]. This approach involves concepts which are also valid for matrices whose elements lie in a finite field, such as the Lagrange interpolation polynomial for a given function and the minimal polynomial of a matrix. Given a transform matrix $\mathbf{M}$, our goal is computing $f(\mathbf{M}) = \mathbf{M}^a$, where the fractional parameter $a = a_1/a_2$ is a ratio of two integers. Compared with previous works concerning fractional NTT [21,33], our proposal has two main advantages. First, it does not require the construction of NTT eigenvector sets, which usually constitutes a laborious task from the point of view of computational complexity. Furthermore, since the fractional transform matrices resulting from the proposed procedure correspond to a linear combination of integer powers of the respective ordinary transform matrices, standard fast algorithms can be straightly employed.

Although the best-known NTT can be viewed as a finite field analogous of the discrete Fourier transform (DFT),[1] other types of NTT have been introduced in recent years. Cosine and Hartley NTT, for example, have interesting properties and can be used in applications related to information hiding, image encryption, communications, etc. [10,12,18,24] In the present work, we also consider such transforms and show how the matrix functions approach can be applied to fractionalize them. In order to avoid ambiguities and contribute to the readability of our text, we adopt the following nomenclature:

FNT	Fourier number-theoretic transform
HNT	Hartley number-theoretic transform
CNT	Cosine number-theoretic transform
SNT	Sine number-theoretic transform
FrFNT	Fractional Fourier number-theoretic transform
FrHNT	Fractional Hartley number-theoretic transform
FrCNT	Fractional cosine number-theoretic transform
FrSNT	Fractional sine number-theoretic transform
$\mathrm{GF}(p)$	Finite field with$p$ elements

This paper is organized as follows. In Sect. 2, we review the main concepts related to trigonometry over finite fields and number-theoretic transforms. In Sect. 3, we develop the NTT fractionalization approach based on matrix functions and use it to define the FrFNT, the FrHNT, the FrCNT, and the FrSNT. The relationship between the FrFNT and the FrHNT is presented in Sect. 4. In Sect. 5, we give some illustrative examples regarding fractional NTT. The image encryption scheme proposed in [19] is revisited in Sect. 6; we show that, using fractional NTT based on matrix functions, its implementation becomes simpler, while its robustness against the main cryptographic attacks remains unaltered. The paper closes with some concluding remarks on Sect. 7.

2 Preliminaries

2.1 Cosine and Sine over Finite Fields

In this section, we present a definition for cosine and sine functions over finite fields. Such finite field trigonometric functions were originally introduced in [7], as a requirement for defining a Hartley number-theoretic transform, and are computed with respect to elements in the set of Gaussian integers over finite fields, $\mathrm{GI}(p)$. This set is isomorphic to the extension field $\mathrm{GF}(p^2)$ and provides an analogy with classical complex numbers.

[1] In an NTT, the kernel $W_N = \mathrm{e}^{-j2\pi/N}$ of an N-point DFT is basically replaced by an element $\zeta \in \mathrm{GF}(p)$ of multiplicative order $\mathrm{ord}(\zeta) = N$, and all computations are carried out modulo p.

Definition 1 The set of Gaussian integers over $\mathrm{GF}(p)$ is the set $\mathrm{GI}(p) := \{c + jd, c, d \in \mathrm{GF}(p)\}$, where j^2 is a quadratic nonresidue over $\mathrm{GF}(p)$.

The elements $\zeta = c + jd$ of the "complex" structure $\mathrm{GI}(p)$ have a "real" part $c = \Re\{\zeta\}$ and an "imaginary" part $c = \Im\{\zeta\}$. If $p \equiv 3 \pmod 4$, one may use $j = \sqrt{-1} \equiv \sqrt{p-1} \pmod p$, for example. However, if $p \equiv 1 \pmod 4$, $-1 \equiv p - 1 \pmod p$ is a quadratic residue and another j has to be selected [6].

Definition 2 Let $\zeta \in \mathrm{GI}(p)$ be an element with multiplicative order denoted by $\mathrm{ord}(\zeta) = N$. The finite field cosine and the finite field sine of the arc related to ζ^x are computed modulo p, respectively, by

$$\cos_\zeta(x) := \frac{\zeta^x + \zeta^{-x}}{2} \tag{1}$$

and

$$\sin_\zeta(x) := \frac{\zeta^x - \zeta^{-x}}{2j}, \tag{2}$$

for $x = 0, 1, \ldots, N - 1$.

2.2 Number-Theoretic Transforms

In what follows, we define Fourier, Hartley, cosine and sine number-theoretic transforms [5,7,20]. Compared with usual NTT definitions, where vectors whose elements lie in $\mathrm{GF}(p)$ are considered, we consider a more general case, where vectors whose elements lie in $\mathrm{GI}(p)$ can be transformed.

Definition 3 Let $\zeta \in \mathrm{GI}(p)$ be an element of multiplicative order $\mathrm{ord}(\zeta) = N$. The Fourier number-theoretic transform (FNT) of an N-length vector $\mathbf{x} = (x[i])$, $x[i] \in \mathrm{GI}(p)$, is an N-length vector $\mathbf{X}_F = (X_F[k])$, $X_F[k] \in \mathrm{GI}(p)$, given by

$$X_F[k] := \sqrt{N^{-1}} \sum_{i=0}^{N-1} x[i]\zeta^{-ki}. \tag{3}$$

The inverse transform is given by

$$x[i] = \sqrt{N^{-1}} \sum_{k=0}^{N-1} X_F[k]\zeta^{ki}.$$

Definition 4 Let $\zeta \in \mathrm{GI}(p)$ be an element of multiplicative order $\mathrm{ord}(\zeta) = N$. The Hartley number-theoretic transform (HNT) of an N-length vector $\mathbf{x} = (x[i])$, $x[i] \in \mathrm{GI}(p)$, is an N-length vector $\mathbf{X}_H = (X_H[k])$, $X_H[k] \in \mathrm{GI}(p)$, given by

$$X_H[k] := \sqrt{N^{-1}} \sum_{i=0}^{N-1} x[i]\mathrm{cas}_\zeta(ki), \tag{4}$$

where $\text{cas}_\zeta(\cdot) := \cos_\zeta(\cdot) + \sin_\zeta(\cdot)$. The inverse transform is also obtained from the expression above, i.e., the Hartley transform is an involution.

The family of trigonometric number-theoretic transforms includes eight types of cosine transforms (CNT) and eight types of sine transforms (SNT) [20]. The construction of a trigonometric NTT is based on symmetric extensions of a sequence whose elements are in a finite field and requires the weighting function

$$\beta[r] = \begin{cases} \sqrt{2^{-1}} \pmod{p}, & r = 0 \text{ or } N, \\ & r = 1, 2, \ldots, N-1. \end{cases}$$

Definition 5 Let $\zeta \in \text{GI}(p)$ be an element of multiplicative order $\text{ord}(\zeta) = 2N$. The cosine number-theoretic transforms of types 1 and 4 (respectively, denoted by CNT-1 and CNT-4) of $(N+1)$- and N-length vectors $\mathbf{x}$ are, respectively, given by

$$X_{C_1}[k] := \sqrt{\frac{2}{N}} \sum_{i=0}^{N} \beta[i]\,\beta[k]\,x[i]\,\cos_\zeta(ki), \tag{5}$$

$k = 0, 1, \ldots, N$, and

$$X_{C_4}[k] := \sqrt{\frac{2}{N}} \sum_{i=0}^{N-1} x[i]\,\cos_\zeta\left(\left(k+\frac{1}{2}\right)\left(i+\frac{1}{2}\right)\right), \tag{6}$$

$k = 0, 1, \ldots, N-1$.

Definition 6 Let $\zeta \in \text{GI}(p)$ be an element of multiplicative order $\text{ord}(\zeta) = 2N$. The sine number-theoretic transforms of types 1 and 4 (respectively, denoted by SNT-1 and SNT-4) of $(N-1)$- and N-length vectors $\mathbf{x}$ are, respectively, given by

$$X_{S_1}[k] := \sqrt{\frac{2}{N}} \sum_{i=1}^{N-1} x[i]\,\sin_\zeta(ki), \tag{7}$$

$k = 1, 2, \ldots, N-1$, and

$$X_{S_4}[k] := \sqrt{\frac{2}{N}} \sum_{i=0}^{N-1} x[i]\,\sin_\zeta\left(\left(k+\frac{1}{2}\right)\left(i+\frac{1}{2}\right)\right), \tag{8}$$

$k = 0, 1, \ldots, N-1$.

We remark that, although an element $\zeta \in \text{GI}(p)$ of multiplicative order $\text{ord}(\zeta) = 2N$ is used to define each CNT and SNT, transforms of different lengths are obtained (N, $N+1$, and $N-1$). This is due to differences among the symmetric extension criteria employed in the construction of each transform type. A complete explanation regarding this aspect can be found in [20,26]. CNT and SNT of types 1 and 4 are also involutions.

2.3 Eigenstructure of Number-Theoretic Transforms

The computation of each transform defined in Sect. 2.2 can be expressed by the matrix equation $\mathbf{X} = \mathbf{Mx}$, where $\mathbf{x}$ is the vector to be transformed, $\mathbf{X}$ is the transform vector, and $\mathbf{M}$ is a transform matrix. In particular, the FNT of a vector $\mathbf{x}$ can be expressed by $\mathbf{X}_F = \mathbf{Fx}$, where the element in the kth row and the ith column of $\mathbf{F}$ is $[\mathbf{F}]_{k,i} = \sqrt{N^{-1}}\zeta^{-ki}$. The inverse of $\mathbf{F}$ is $\mathbf{F}^{-1} = \mathbf{PF}$, where $\mathbf{P}$ is an $N \times N$ Schur matrix [4] given by

$$\mathbf{P} = \begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \mathbf{J} \end{bmatrix}, \tag{9}$$

and $\mathbf{J}$ is an $(N-1) \times (N-1)$ matrix with ones in the antidiagonal.

The matrix $\mathbf{F}$ has period equal to 4, i.e., $l = 4$ is the least positive integer such that $\mathbf{F}^l = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. On the other hand, the matrices related to the HNT, the CNT-1, the CNT-4, the SNT-1, and the SNT-4, respectively, denoted by $\mathbf{H}$, $\mathbf{C}_1$, $\mathbf{C}_4$, $\mathbf{S}_1$, and $\mathbf{S}_4$, are unitary and also symmetric. This means that such matrices have period equal to 2 [7,24]. These facts can be used to determine the eigenvalues of each transform matrix as well as their multiplicities. Such results are summarized in the following proposition [4,24,31].

Proposition 1 *The eigenvalues of an NTT matrix are distributed as follows:*

- *The matrix* $\mathbf{F}$ *has, at most, four distinct eigenvalues, namely* $\left\{1, -1, \sqrt{-1}, -\sqrt{-1}\right\}$, *whose multiplicities are shown in Table 1;*
- *The matrix* $\mathbf{H}$ *has, at most, two distinct eigenvalues, namely* $\{1, -1\}$, *whose multiplicities are shown in Table 1;*
- *The matrices* $\mathbf{C}_1$, $\mathbf{C}_4$, $\mathbf{S}_1$, *and* $\mathbf{S}_4$ *have, at most, two distinct eigenvalues, namely* $\{1, -1\}$, *whose multiplicities are shown in Table 2.*

Table 1 Multiplicities of the eigenvalues of $N \times N$ matrices of the FNT and the HNT

N	Fourier				Hartley	
	1	-1	$\sqrt{-1}$	$-\sqrt{-1}$	1	-1
$4n$	$n+1$	n	$n-1$	n	$2n+1$	$2n-1$
$4n+1$	$n+1$	n	n	n	$2n+1$	$2n$
$4n+2$	$n+1$	$n+1$	n	n	$2n+1$	$2n+1$
$4n+3$	$n+1$	$n+1$	n	$n+1$	$2n+2$	$2n+1$

Table 2 Multiplicities of the eigenvalues of $N' \times N'$ matrices (N' can be equal to N, $N-1$ or $N+1$) of the CNT and the SNT of types 1 and 4

N'	1	-1
Odd	$\frac{N'+1}{2}$	$\frac{N'-1}{2}$
Even	$\frac{N'}{2}$	$\frac{N'}{2}$

3 Finite Field Fractional Transforms Based on Matrix Functions

The approach presented in this section is based on matrix functions [14], whose theory can be described using concepts which are valid also in the finite field scenario. Such concepts include, for example, the Lagrange interpolating polynomial for a given function and the minimal polynomial of a matrix [17]. Our goal is to compute the function $\mathbf{A}^a$, where $\mathbf{A}$ is an NTT matrix and a is a rational number called *fractional parameter*. We start by considering the minimal polynomial of $\mathbf{A}$, which is defined as the unique monic polynomial ψ of lowest degree such that $\psi(\mathbf{A}) = 0$. The minimal polynomial divides any other polynomial r for which $r(\mathbf{A}) = 0$. According to the following theorem, $r(\mathbf{A})$ is completely determined by the values of r on the spectrum of $\mathbf{A}$.

Theorem 1 *Let r and s be polynomials and $\mathbf{A}$ be an $N \times N$ matrix over a finite field. Then $r(\mathbf{A}) = s(\mathbf{A})$ if and only if r and s take the same values on the spectrum of $\mathbf{A}$.*

The proof of Theorem 1 is analogous to that presented for Theorem 1.3 on p. 5 of [14]. This result can be generalized to an arbitrary function f considering the following definitions.

Definition 7 Let $\lambda_1, \lambda_2, \ldots, \lambda_v$ be the distinct eigenvalues of $\mathbf{A}$, and let n_i be the dimension of the largest Jordan block in which λ_i appears. We say that $f(t)$ is defined on the spectrum of $\mathbf{A}$ if the kth derivatives

$$f^{(k)}(\lambda_i), \quad k = 0, 1, \ldots, n_i - 1, \quad i = 1, \ldots, v, \tag{10}$$

called the values of $f(t)$ on the spectrum of $\mathbf{A}$, exist.

Definition 8 Let f be a polynomial defined on the spectrum of an $N \times N$ matrix $\mathbf{A}$ over a finite field, and let ψ be the minimal polynomial of $\mathbf{A}$. Then $f(\mathbf{A}) = r(\mathbf{A})$, where r is the polynomial of degree less than $\deg(\psi)$ that satisfies the interpolation condition

$$r^{(k)}(\lambda_i) = f^{(k)}(\lambda_i), \tag{11}$$

$k = 0, 1, \ldots, n_i - 1$, $i = 1, 2, \ldots, v$. If $n_i = 1, i = 1, \ldots, v$, the polynomial r corresponds to the Lagrange interpolating polynomial [17]

$$r(t) = \sum_{i=1}^{v} f(\lambda_i) l_i(t), \tag{12}$$

where

$$l_i(t) = \prod_{j=1,\, j \neq i}^{v} \frac{t - \lambda_j}{\lambda_i - \lambda_j}. \tag{13}$$

In order to use Definition 8 to compute the *ath* power of a transform matrix, we set $r(t) = t^a$, where $a = a_1/a_2, a_2 \neq 0$, is a ratio of two integers; the variable t is replaced by the corresponding NTT matrix. In what follows, we give details related to the development of this approach to each NTT defined in Sect. 2.

3.1 Fractional Fourier Number-Theoretic Transform

The fractional Fourier number-theoretic transform (FrFNT) of an N-length vector $\mathbf{x}$ is computed as $\mathbf{X}_{a,F} = \mathbf{F}^a\mathbf{x}$. According to Proposition 1, the FNT matrix has four distinct eigenvalues $\lambda \in \{1, -1, \sqrt{-1}, -\sqrt{-1}\}$ and, therefore, $v = 4$. Thus, the functions $l_i(t)$, $i = 0, 1, 2, 3$, defined in Eq. (13), are

$$\begin{aligned} l_1(t) &= \frac{t^3 + t^2 + t + 1}{4}, \\ l_2(t) &= \frac{-t^3 + t^2 - t + 1}{4}, \\ l_3(t) &= \frac{\sqrt{-1}\,t^3 - t^2 - \sqrt{-1}\,t + 1}{4} \end{aligned}$$

and

$$l_4(t) = \frac{-\sqrt{-1}\,t^3 - t^2 + \sqrt{-1}\,t + 1}{4}.$$

From Eq. (12), one has

$$r(t) = l_1(t) + (-1)^a l_2(t) + (\sqrt{-1})^a l_3(t) + (-\sqrt{-1})^a l_4(t).$$

Grouping the coefficients of each power of t and setting $\alpha_i(a) = \alpha_i(a_1, a_2)$ as the coefficient of the ith power of t, one has

$$r(t) = \sum_{i=0}^{3} \alpha_i(a_1, a_2) t^i,$$

where

$$\begin{aligned} \alpha_0(a_1, a_2) &= \frac{1 + (\sqrt{-1})^a + (-1)^a + (-\sqrt{-1})^a}{4} \\ &= \frac{1}{4}\left[(\sqrt[2a_2]{-1})^{a_1}\left(1 + (\sqrt[2a_a]{-1})^{a_1}\right) + (\sqrt[2a_2]{-1})^{-a_1}\left(1 + (\sqrt[2a_a]{-1})^{a_1}\right)\right] \\ &= \frac{1 + (\sqrt[2a_2]{-1})^{a_1}}{2} \cos_{\sqrt[2a_2]{-1}}(a_1), \qquad (14) \\ \alpha_1(a_1, a_2) &= \frac{1 - \sqrt{-1}(\sqrt[2a_2]{-1})^{a_1}}{2} \sin_{\sqrt[2a_2]{-1}}(a_1), \qquad (15) \end{aligned}$$

$$\alpha_2(a_1, a_2) = \frac{-1 + (\sqrt[2a_2]{-1})^{a_1}}{2} \cos_{\sqrt[2a_2]{-1}}(a_1), \tag{16}$$

$$\alpha_3(a_1, a_2) = \frac{-1 - \sqrt{-1}(\sqrt[2a_2]{-1})^{a_1}}{2} \sin_{\sqrt[2a_2]{-1}}(a_1). \tag{17}$$

Finally, the FrFNT matrix with fractional parameter $a = a_1/a_2$ is computed as

$$\mathbf{F}^a = \mathbf{F}^{\frac{a_1}{a_2}} = r(\mathbf{F}) = \sum_{i=0}^{3} \alpha_i(a_1, a_2)\mathbf{F}^i. \tag{18}$$

3.2 Fractional Hartley, Cosine and Sine Number-Theoretic Transforms

In order to define fractional Hartley (FrHNT), cosine of type 1 (FrCNT-1) and type 4 (FrCNT-4), sine of type 1 (FrSNT-1) and type 4 (FrSNT-4) number-theoretic transforms, we compute $\mathbf{H}^a$, $\mathbf{C}_1^a$, $\mathbf{C}_4^a$, $\mathbf{S}_1^a$ and $\mathbf{S}_4^a$, respectively. According to Proposition 1, these matrices have two distinct eigenvalues $\lambda \in \{1, -1\}$. In these cases, the functions $l_i(t)$, $i = 0, 1$, defined in Eq. (13), are

$$l_1(t) = \frac{1+t}{2}$$

and

$$l_2(t) = \frac{1-t}{2}.$$

From Eq. (12), one has

$$r(t) = \left(\frac{1+t}{2}\right) + (-1)^a \left(\frac{1-t}{2}\right). \tag{19}$$

Grouping the coefficients of the powers of t and setting $\alpha_i(a) = \alpha_i(a_1, a_2)$ as the coefficient of the ith power of t, one has

$$r(t) = \alpha_0(a_1, a_2) + \alpha_1(a_1, a_2)t, \tag{20}$$

where

$$\alpha_0(a_1, a_2) = \frac{1 + (-1)^a}{2}$$

and

$$\alpha_1(a_1, a_2) = \frac{1 - (-1)^a}{2}.$$

Finally, the fractional number-theoretic transform matrix $\mathbf{B}^a$ with fractional parameter $a = (a_1/a_2)$ is computed as

$$\mathbf{B}^a = \alpha_0(a_1, a_2)\mathbf{I} + \alpha_1(a_1, a_2)\mathbf{B}, \tag{21}$$

where $\mathbf{B}$ can be replaced by $\mathbf{H}$, $\mathbf{C}_1$, $\mathbf{C}_4$, $\mathbf{S}_1$ or $\mathbf{S}_4$, according to the fractional transform one desires to compute.

We reinforce the fact that the computation of fractional powers of an NTT matrix by means of matrix functions does not require the construction of eigenvector sets. This makes our approach simpler than that employed in [21,22,33], where fractional Fourier, cosine and sine number-theoretic transforms are defined via finite field extensions of the methods given in [8,32,34]. In particular, the approach described in [21] involves a systematic procedure for deriving finite field Hermite–Gaussian vectors, whose components may unpredictably lie in higher-order extension fields. On the other hand, using the technique proposed in the present paper, if $\sqrt[2a_2]{-1} \in \mathrm{GI}(p)$, we assure that the components of $\mathbf{F}^a = \mathbf{F}^{\frac{a_1}{a_2}}$ lie in $\mathrm{GI}(p)$; correspondingly, for other transforms, it is sufficient that $\sqrt[a_2]{-1} \in \mathrm{GI}(p)$.

3.3 Fast Algorithms

Another important property of fractional NTT defined using matrix functions is that they can be computed by means of standard fast algorithms. This is due to the fact that, according to the referred approach, the ath power of a transform matrix $\mathbf{B}$ is a linear combination of integer powers of $\mathbf{B}$. To be more specific, let us consider an N-point Fourier number-theoretic transform, for which the matrix-vector product $\mathbf{Fx}$ can be computed by means of a fast algorithm with $M_F(N)$ multiplications and $A_F(N)$ additions. According to Eq. (18), the FrFNT with fractional parameter $a = a_1/a_2$ of an N-point vector $\mathbf{x}$ can be computed as

$$\begin{aligned}
\mathbf{X}_{a,F} &= \left[\alpha_0(a_1,a_2)\mathbf{I} + \alpha_1(a_1,a_2)\mathbf{F} + \alpha_2(a_1,a_2)\mathbf{F}^2 + \alpha_3(a_1,a_2)\mathbf{F}^3\right]\mathbf{x} \\
&= \left[\alpha_0(a_1,a_2)\mathbf{I} + \alpha_1(a_1,a_2)\mathbf{F} + \alpha_2(a_1,a_2)\mathbf{P} + \alpha_3(a_1,a_2)\mathbf{PF}\right]\mathbf{x} \\
&= \left[\alpha_0(a_1,a_2)\mathbf{I} + \alpha_2(a_1,a_2)\mathbf{P}\right]\mathbf{x} + \left[\alpha_1(a_1,a_2)\mathbf{I} + \alpha_3(a_1,a_2)\mathbf{P}\right]\mathbf{Fx}.
\end{aligned}$$

By observing the last equation, we see that the FrFNT requires $M'_F(N) = M_F(N) + 2[2N - 2 + N \pmod 2)]$ multiplications and $A'_F(N) = A_F(N) + 3N$ additions. Similarly, let us consider an N-point HNT, CNT, or SNT, for which the matrix-vector product $\mathbf{Bx}$ can be computed by means of a fast algorithm with $M_B(N)$ multiplications and $A_B(N)$ additions. According to Eq. (21), the FrHNT, the FrCNT, or the FrSNT with fractional parameter $a = a_1/a_2$ of an N-point vector $\mathbf{x}$ can be computed as

$$\mathbf{X}_{a,B} = \left[\alpha_0(a_1,a_2)\mathbf{I} + \alpha_1(a_1,a_2)\mathbf{B}\right]\mathbf{x} = \alpha_0(a_1,a_2)\mathbf{x} + \alpha_1(a_1,a_2)\mathbf{Bx}.$$

Therefore, the computation of $\mathbf{X}_{a,B}$ involves $M'_{\mathbf{B}}(N) = M_B(N) + 2N$ multiplications and $A'_B(N) = A_B(N) + N$ additions.

In general, fast algorithms applicable to real-valued discrete transforms can also be used to compute number-theoretic transforms. In this sense, FNT, HNT and, consequently, FrFNT and FrHNT, can be computed by means of Cooley–Tukey, Good–Thomas prime factor, and Rader prime algorithms, for instance [5,35]. Similarly, since the CNT and the SNT matrices have symmetries analogous to those of the corresponding real-valued transforms, standard decimation-in-time and decimation-in-frequency fast algorithms can be employed in their computation and also in the computation of the respective fractional number-theoretic transforms [9,15].

In short, an N-point fractional NTT based on matrix functions can be computed with $\mathcal{O}(N \log N)$ additions and multiplications in the corresponding field; on the other hand, the computation of the N-point fractional NTT defined in [21,22], for example, requires $\mathcal{O}(N^2)$ multiplications and additions. Moreover, if the transform is defined over fields whose characteristic is a Fermat or a Mersenne prime, other strategies can be employed to further decrease the computational complexity. Such strategies include the employment of residue number system and the implementation of multiplications by means of bit-shifts [5].

4 Relationship Between the FrFNT and the FrHNT

In this section, we show that the FrFNT matrix can be obtained from the FrHNT matrix and vice versa. The relationship between such two matrices is a generalization of the relationship between the FNT and the HNT matrices. For the sake of simplicity, in what follows, we assume that $p \equiv 3 \pmod 4$ and $j = \sqrt{-1}$. From Definitions 3 and 4, one clearly observes that matrices $\mathbf{F}$ and $\mathbf{H}$ are related according to

$$\mathbf{F} = \frac{1}{2}(\mathbf{I} + \mathbf{P})\mathbf{H} + \frac{j}{2}(\mathbf{I} - \mathbf{P})\mathbf{H} \tag{22}$$

or

$$\mathbf{H} = \left[\frac{1}{2}(1 - j)\mathbf{I} + \frac{1}{2}(1 + j)\mathbf{P}\right]\mathbf{F}. \tag{23}$$

Defining the matrix $\mathbf{S}$ as

$$\mathbf{S} := \frac{1}{2}(1 - j)\mathbf{I} + \frac{1}{2}(1 + j)\mathbf{P},$$

we may write $\mathbf{H} = \mathbf{SF}$ and $\mathbf{F} = \mathbf{S}^{-1}\mathbf{H}$. If N is odd, the $N \times N$ matrix $\mathbf{S}$ is

$$\mathbf{S} = \frac{1}{2}\begin{bmatrix} 2 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 1-j & \cdots & 0 & 0 & \cdots & 1+j \\ \vdots & \vdots & \ddots & \vdots & \vdots & ⋰ & \vdots \\ 0 & 0 & \cdots & 1-j & 1+j & \cdots & 0 \\ 0 & 0 & \cdots & 1+j & 1-j & \cdots & 0 \\ \vdots & \vdots & ⋰ & \vdots & \vdots & \ddots & \vdots \\ 0 & 1+j & \cdots & 0 & 0 & \cdots & 1-j \end{bmatrix};$$

if N is even, the $N \times N$ matrix $\mathbf{S}$ has the form

$$\mathbf{S} = \frac{1}{2}\begin{bmatrix} 2 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 1-j & \dots & 0 & 0 & 0 & \dots & 1+j \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \iddots & \vdots \\ 0 & 0 & \dots & 1-j & 0 & 1+j & \dots & 0 \\ 0 & 0 & \dots & 0 & 2 & 0 & \dots & 0 \\ 0 & 0 & \dots & 1+j & 0 & 1-j & \dots & 0 \\ \vdots & \vdots & \iddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1+j & \dots & 0 & 0 & 0 & \dots & 1-j \end{bmatrix}.$$

Our goal is to find $\mathbf{S}^a$, which can be done by using matrix functions. We begin by characterizing $\mathbf{S}$ with respect to its eigenstructure. The proofs of the following results are given in the "Appendix."

Lemma 1 *The matrix* $\mathbf{S}$ *has, at most, four distinct eigenvalues.*

Lemma 2 *The* $N \times N$ *matrix*

$$\mathbf{B} = \begin{bmatrix} b & 0 & \dots & 0 & 0 & \dots & 0 & c \\ 0 & b & \dots & 0 & 0 & \dots & c & b \\ \vdots & \vdots & \ddots & \vdots & \vdots & \iddots & \vdots & 0 \\ 0 & 0 & \dots & b & c & \dots & 0 & 0 \\ 0 & 0 & \dots & c & b & \dots & 0 & 0 \\ \vdots & \vdots & \iddots & \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & c & \dots & 0 & 0 & \dots & b & 0 \\ c & 0 & \dots & 0 & 0 & \dots & 0 & b \end{bmatrix} \tag{24}$$

has determinant $\left(b^2 - c^2\right)^{\frac{N}{2}}$.

Lemma 3 *If* N *is odd, the minimal polynomial of* $\mathbf{S}$ *is*

$$P_{\mathbf{S}}(\lambda) = (\lambda - 1)\left[(2\lambda - 1 + j)^2 - (1 + j)^2\right]^{\frac{N-1}{2}}.$$

If N *is even, the minimal polynomial of* $\mathbf{S}$ *is*

$$P_{\mathbf{S}}(\lambda) = (\lambda - 1)^2\left[(2\lambda - 1 + j)^2 - (1 + j)^2\right]^{\frac{N-2}{2}}.$$

Theorem 2 *The eigenvalues of the matrix* $\mathbf{S}$ *are* $\{1, -j\}$. *Their multiplicities are shown in* Table 3.

Since the matrix $\mathbf{S}$ has two distinct eigenvalues, and according to Definition 8, its $l_i(t)$ functions are

$$l_1(t) = \frac{t + j}{1 + j} = \frac{1 - j}{2}(j + t)$$

Table 3 Multiplicities of the eigenvalues of the $N \times N$ matrix **S**

N	1	$-j$
Even	$\frac{N+2}{2}$	$\frac{N-2}{2}$
Odd	$\frac{N+1}{2}$	$\frac{N-1}{2}$

and

$$l_2(t) = \frac{t-1}{-j-1} = \frac{1-j}{2}\,(1-t)\,,$$

the Lagrange interpolating polynomial is

$$r(t) = (1)^a \frac{1-j}{2}\,(j+t) + (-j)^a \frac{1-j}{2}\,(1-t)\,.$$

From the definition of the matrix $\mathbf{S}$, the term $\frac{1-j}{2}\,(j\mathbf{I}+\mathbf{S})$ is equivalent to $\frac{\mathbf{I}+\mathbf{P}}{2}$ and the term $\frac{1-j}{2}\,(\mathbf{I}-\mathbf{S})$ is equivalent to $\frac{\mathbf{I}-\mathbf{P}}{2}$. Therefore, replacing t by the matrix $\mathbf{S}$, one obtains

$$\mathbf{S}^a = r(\mathbf{S}) = \frac{1}{2}(\mathbf{I}+\mathbf{P}) + \frac{(-j)^a}{2}(\mathbf{I}-\mathbf{P})$$

and the relationship between the FrHNT and the FrFNT, which is given by

$$\mathbf{H}^a = \left[\frac{1}{2}(\mathbf{I}+\mathbf{P}) + \frac{(-j)^a}{2}(\mathbf{I}-\mathbf{P})\right]\mathbf{F}^a. \qquad (25)$$

5 Examples

In this section, we develop numerical examples to illustrate the construction of fractional number-theoretic transforms based on matrix functions.

5.1 FrFNT

Let us consider the element $\zeta = 4 \in \mathrm{GI}(257)$, where $\mathrm{ord}(4) = 8$. We choose the fractional parameter $a = a_1/a_2 = 3/8$ and use Definition 8 to construct an 8×8 FrFNT matrix. Initially, we obtain the 8×8 FNT matrix

$$\mathbf{F} = \begin{bmatrix} 242 & 242 & 242 & 242 & 242 & 242 & 242 & 242 \\ 242 & 197 & 17 & 68 & 15 & 60 & 240 & 189 \\ 242 & 17 & 15 & 240 & 242 & 17 & 15 & 240 \\ 242 & 68 & 240 & 197 & 15 & 189 & 17 & 60 \\ 242 & 15 & 242 & 15 & 242 & 15 & 242 & 15 \\ 242 & 60 & 17 & 189 & 15 & 197 & 240 & 68 \\ 242 & 240 & 15 & 17 & 242 & 240 & 15 & 17 \\ 242 & 189 & 240 & 60 & 15 & 68 & 17 & 197 \end{bmatrix}$$

using Definition 3. In order to obtain $\alpha_i(a_1, a_2)$, $i = 0, 1, 2, 3$, we compute

$$\begin{aligned}\sqrt[16]{(-1)^3} \equiv \sqrt[16]{(256)^3} \equiv 60^3 \equiv 120 \pmod{257},\\ \cos_{\sqrt[16]{-1}}(3) = \cos_{60}(3) = 196,\\ \sin_{\sqrt[16]{-1}}(3) = \sin_{60}(3) = 188.\end{aligned}$$

Using these results, one obtains $\alpha_0(3,8) = 36$, $\alpha_1(3,8) = 28$, $\alpha_2(3,8) = 97$, $\alpha_3(3,8) = 97$ and, therefore,

$$\mathbf{F}^{\frac{3}{8}} = 36\mathbf{F}^0 + 28\mathbf{F}^1 + 97\mathbf{F}^2 + 97\mathbf{F}^3 = \begin{bmatrix} 57 & 181 & 181 & 181 & 181 & 181 & 181 & 181 \\ 181 & 241 & 112 & 14 & 76 & 52 & 145 & 83 \\ 181 & 112 & 112 & 145 & 181 & 112 & 173 & 145 \\ 181 & 14 & 145 & 241 & 76 & 83 & 112 & 52 \\ 181 & 76 & 181 & 76 & 57 & 76 & 181 & 76 \\ 181 & 52 & 112 & 83 & 76 & 241 & 145 & 14 \\ 181 & 145 & 173 & 112 & 181 & 145 & 112 & 112 \\ 181 & 83 & 145 & 52 & 76 & 14 & 112 & 241 \end{bmatrix}.$$

5.2 FrHNT

In this example, we consider the same parameters of Example 5.1 and construct the 8×8 FrHNT matrix using Eq. (20). From Definition 4, the 8×8 HNT matrix shown in Eq. (26) is constructed. In order to obtain $\alpha_i(a_1, a_2)$, $i = 0, 1$, we compute $\sqrt[8]{-1} \equiv 2 \pmod{257}$, $\alpha_0(3,8) = 133$ and $\alpha_1(3,8) = 125$. The matrix $\mathbf{H}^{3/8} = 133\mathbf{H}^0 + 125\mathbf{H}^1$, shown in Eq. (27), is obtained.

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 227+223j & 241j & 30+223j & 255 & 228 & 130 & 103 \\ 1 & 241j & 255 & 16j & 103 & 1 & 29 & 178 \\ 1 & 30+223j & 16j & 227+223j & 67 & 255 & 130 & 103 \\ 1 & 255 & 1 & 255 & 1 & 1 & 1 & 1 \\ 1 & 30+223j & 241j & 227+34j & 67 & 255 & 130 & 103 \\ 1 & 16j & 255 & 241j & 103 & 1 & 29 & 178 \\ 1 & 227+223j & 16j & 30+34j & 67 & 255 & 130 & 103 \end{bmatrix} \tag{26}$$

$$\mathbf{H}^{\frac{3}{8}} = \begin{bmatrix} 1 & 125 & 125 & 125 & 125 & 125 & 125 & 125 \\ 125 & 238+118j & 56j & 152+119j & 132 & 152+138j & 201j & 105+138j \\ 125 & 56j & 8 & 201j & 125 & 56j & 132 & 201j \\ 125 & 152+119j & 201j & 238+119j & 132 & 105+138j & 56j & 152+138j \\ 125 & 132 & 125 & 132 & 1 & 132 & 125 & 132 \\ 125 & 152+138j & 56j & 105+138j & 132 & 238+119j & 201j & 152+119j \\ 125 & 201j & 132 & 56j & 125 & 201j & 8 & 56j \\ 125 & 105+138j & 201j & 152+138j & 132 & 152+119j & 56j & 238+119j \end{bmatrix} \tag{27}$$

5.3 FrCNT

In order to obtain the 8×8 FrCNT of type 4 matrix over GI(257), an element ζ whose multiplicative order is $\mathrm{ord}(\zeta) = 2N = 16$ has to be chosen. We then select the element $\zeta = 2 \in \mathrm{GI}(257)$, which has the mentioned order, and use again the fractional parameter $a = a_1/a_2 = 3/8$. From Eq. (6), the 8×8 CNT-4 matrix

$$\mathbf{C}_4 = \begin{bmatrix} 29 & 11 & 190 & 127 & 189 & 178 & 154 & 61 \\ 11 & 189 & 61 & 79 & 67 & 228 & 130 & 103 \\ 190 & 61 & 130 & 246 & 103 & 189 & 29 & 178 \\ 127 & 79 & 246 & 61 & 29 & 154 & 67 & 68 \\ 189 & 67 & 103 & 29 & 196 & 246 & 178 & 127 \\ 178 & 228 & 189 & 154 & 246 & 127 & 61 & 67 \\ 154 & 130 & 29 & 67 & 178 & 61 & 68 & 11 \\ 61 & 103 & 178 & 68 & 127 & 67 & 11 & 228 \end{bmatrix}$$

is constructed. We use Eq. (20) with $\alpha_0(3, 8) = 133$ and $\alpha_1(3, 8) = 125$ (the same values employed in the construction of the FrHNT matrix in Example 5.2). Thus, one has

$$\mathbf{C}_4^{\frac{3}{8}} = 133\mathbf{C}_4^0 + 125\mathbf{C}_4^1 = \begin{bmatrix} 160 & 90 & 106 & 198 & 238 & 148 & 232 & 172 \\ 90 & 114 & 172 & 109 & 151 & 230 & 59 & 25 \\ 106 & 172 & 192 & 167 & 25 & 238 & 27 & 148 \\ 198 & 109 & 167 & 48 & 27 & 232 & 151 & 19 \\ 238 & 151 & 25 & 27 & 218 & 167 & 148 & 198 \\ 148 & 230 & 238 & 232 & 167 & 74 & 172 & 151 \\ 232 & 59 & 27 & 151 & 148 & 172 & 152 & 90 \\ 172 & 25 & 148 & 19 & 198 & 151 & 90 & 106 \end{bmatrix}.$$

The FrCNT of type 1 and FrSNT of types 1 and 4 are constructed in a similar manner.

6 An Image Encryption Scheme Based on Fractional NTT

In this section, we revisit the image encryption scheme proposed in [19]. More specifically, we verify that, after changing the FrFNT based on the approach given in [21] by the FrFNT constructed using the matrix functions approach, the robustness of the referred scheme against the main cryptographic attacks is not affected. Moreover, we show that the FrFNT we use to encrypt a grayscale image encoded with 8 bpp does not require a *pixel juxtaposition* strategy performed in [19]. In addition to the computational advantages described in Sect. 3.3, this makes the current proposal more efficient and straight.

6.1 Encryption Scheme

The encryption procedure consists in taking $N \times N$ image blocks from left to right and from top to bottom, in the manner shown in Fig. 1. A two-dimensional version of an FrFNT is applied to the current image block and the resulting block replaces the corresponding original block before the next block is processed. In order to provide diffusion to the scheme, a number of columns and rows is shared among adjacent blocks. To be more specific, in Fig. 1, the first o_v columns of block $\mathbf{b}_1$ (before its encryption) correspond to the last o_v columns of the encrypted version of $\mathbf{b}_0$; $\mathbf{b}_1$ is then encrypted, and the first o_v columns of block $\mathbf{b}_2$ (before its encryption) correspond to the last o_v columns of the encrypted version of $\mathbf{b}_1$; $\mathbf{b}_2$ is then encrypted and so on. A similar overlapping strategy is also performed in the rows. An image block may have to be assembled from two sub-blocks; this is the case of $\mathbf{b}_3 = [\mathbf{b}_3^{(1)}|\mathbf{b}_3^{(2)}]$ in Fig. 1.

The fractional parameter used in the computation of the FrFNT of each image block is obtained from a secret-key. Actually, an integer a_2 is chosen and kept fixed. The secret-key is the K-length vector of integers

$$\mathbf{k} = \{a_{1,0}, a_{1,1}, a_{1,2}, \ldots, a_{1,K-2}, a_{1,K-1}\}$$

and the ith image block is transformed by the matrix

$$\mathbf{F}^{\frac{a_{1,i \ (\mathrm{mod}\ K)}}{a_2}}.$$

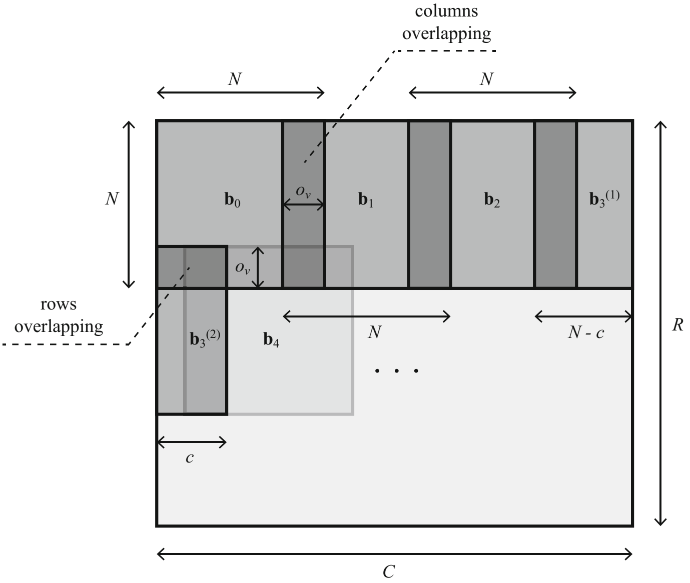

Fig. 1 Block selection and overlapping in the image encryption scheme. The image has dimensions $R \times C$, and blocks with dimensions $N \times N$ are selected. *Darker gray* regions correspond to regions where adjacent blocks overlap

In the computation of the fractional power of $\mathbf{F}$, the index i of the component of $\mathbf{k}$ is taken modulo K because the number of blocks to be processed is usually greater than the key length. In other words, the key components are used in a cyclic manner. The encryption is finished after the whole image is covered by two rounds of the block-by-block transformation procedure we have explained. The decryption consists in applying, in the reverse order, the same steps performed in the encryption.

Specific parameters for an image encryption scheme following the described approach can be chosen in accordance with the type of images to be processed. Throughout this paper, we consider grayscale images encoded with 8 bpp. Since the pixels values range from 0 to 255, an FrFNT defined over GI(257) can be used to encrypt such images. We then consider the FNT used to construct the FrFNT in Example 5.1 and use it to construct FrFNT with different fractional parameters. In this case, image blocks with dimension 8 × 8 are processed.

We remark that, using the method proposed in [21], an 8 × 8 FrFNT matrix over GI(257) would have its entries lying in higher extension fields. Since this would require a more sophisticated arithmetic, in [19], the authors defined a 4 × 4 FrFNT matrix over GI(65537) and applied it to blocks where each entry is a Gaussian integer whose "real" and "imaginary" parts are 16-bit numbers obtained from the juxtaposition of two 8-bit numbers (pixels of an 8 × 8 image block). Using the matrix functions approach, as shown in Example 5.1, 8 × 8 FrFNT matrices whose elements lie in GF(257) can be obtained and any pixel manipulation is unnecessary.

6.2 Computer Experiments and Security Aspects

In order to carry out computer simulations of the proposed scheme, we consider the FrNTT over GI(257) mentioned at the end of Sect. 6.1, and set $a_2 = 64$ and $o_v = 2$ (number of columns and rows overlapping). The secret-key

$$\mathbf{k} = [53, 58, 9, 59, 41, 7, 18, 36, 62, 62, 11, 63, 62, 32, 52, 10, 27, 59, 51, 62, 42, 3, 55, 60, 44, 49, 48, 26, 42, 11, 46, 3, 18, 3, 7, 53, 45, 21, 61, 3]$$

was randomly generated and the 512 × 512 image lena.bmp shown in Fig. 2a was encrypted.

In the encryption procedure, an additional strategy is employed, with the purpose of avoiding the appearance of encrypted image blocks with pixels whose value is 256; the representation of such pixels would require a 9-bit encoding. The strategy consists in applying to the ith image block the FrNTT with fractional parameter $\frac{a_{1,i \pmod{40}}}{64}$ in a recursive manner. More precisely, the FrNTT is applied to the image block and, if the result contains a pixel of value 256, the FrNTT is applied again. The recursive FrNTT application stops when a transformed image block whose maximum pixel value is less than 256 is obtained, which allows maintaining an 8 bpp encoding.

In Fig. 2b, we observe that the visual aspect of the encrypted image is completely noisy. In contrast to the histogram of the original image (Fig. 2c), the histogram of the encrypted image appears to be uniform (Fig. 2d). Additionally, the correlation coefficients obtained from pairs of horizontally adjacent pixels in the original and

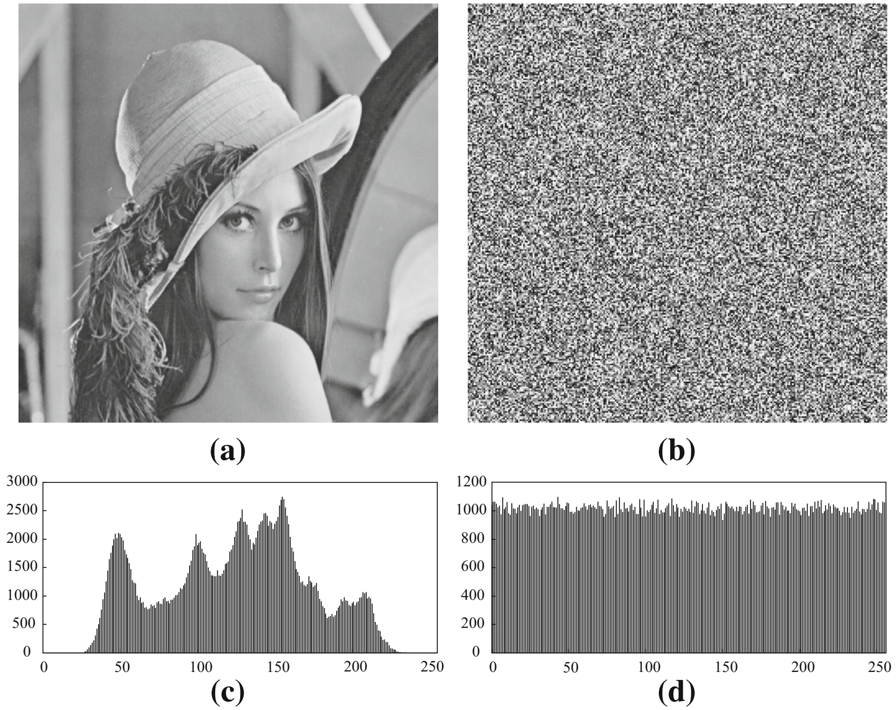

Fig. 2 Image encryption by means of FrNTT. **a** Original image *lena.bmp*; **c** encrypted version of *lena.bmp*; **b** histogram of original image and **d** histogram of encrypted image

the encrypted images are, respectively, 0.9853 and 0.0001 [40]; the entropy of the encrypted image is 7.9992. This suggests that statistical and entropy attacks against the proposed scheme would not be effective.

The secret-key used in our scheme is a vector with 40 integer numbers in the range 1–64. Since each such integer is encoded as a 6-bit string, the key length is 240 bits. This satisfies the general requirement for resisting brute-force attack [38]. Naturally, since the working premises of the scheme do not depend on the key length, the key space size can be easily increased according to the desired security level.

The resistance of the method to differential attack can be measured by the *number of pixels change rate* (NPCR) and the *unified average changing intensity* (UACI), whose ideal values are, respectively, 100% and $33.\overline{3}\%$ [1]. In order to obtain such metrics, we change the least significant bit of a randomly chosen pixel in the original image lena.bmp. The modified image is then encrypted and compared with the encrypted version of the original image. After performing this experiment 100 times, the average NPCR and UACI were, respectively, 99.6083 and 33.4765%. The same procedure was performed for lena.bmp with other resolutions. The results were, respectively, 99.6155 and 33.4300 % for an 128 × 128 image, 99.6307 and 33.4425 % for a 256 × 256 image, and 99.6065 and 33.4565 % for an 1024 × 1024 image. This indicates that the scheme is also robust against differential attack.

We can also perform a preliminary analysis regarding the computational complexity of the proposed encryption scheme. Since the method basically involves transform computations, we characterize its complexity by means of the total number of arithmetic operations needed to apply such transforms. Considering the overlapping of two columns and two rows in the processing of each 8 × 8 image block, we estimate the number of FrNTT necessary to cover the whole $R \times C$ image twice as

$$T_{N=8,\mathrm{GF}(257)} = 2 \times \frac{R}{6} \times \frac{C}{6}.$$

Using the definition proposed in this paper, the computation of an N-point FrNTT requires $\mathcal{O}(N \log N)$ arithmetic operations (see Sect. 3.3). Therefore, assuming that an $N \times N$ two-dimensional FrNTT corresponds to an N-point FrNTT calculated $2N$ times, we estimate the total complexity of our method by

$$16 \times 8 \log 8 \times T_{N=8,\mathrm{GF}(257)} = \frac{512}{24} \times R \times C. \tag{28}$$

The complexity of our method can be compared with that of the method proposed in [19], which employs 4-point FrNTT over GI(65537), defined according to [21].[2] In this case, the number of FrNTT necessary to cover the whole $R \times C$ image twice is estimated as

$$T_{N=4,\mathrm{GI}(65537)} = 2 \times \frac{R}{8} \times \frac{C}{7}.$$

The computation of an N-point FrNTT defined according to [21] requires $\mathcal{O}(N^2)$ arithmetic operations. Since these operations have to be carried out over GI(65537), it is reasonable to consider that their cost is at least 16 times the cost of an operation carried out over GF(257). Thus, we estimate the total complexity of the method given in [19] by

$$8 \times 4^2 \times 4^2 \times T_{N=4,\mathrm{GI}(65537)} = \frac{512}{7} \times R \times C. \tag{29}$$

Comparing Eqs. (28) and (29), one concludes that the computational complexity of the proposed scheme is about 3.4 times less than that of the method given in [19].

Finally, it is important to remark that the image encryption scheme analyzed in this section is very flexible. Instead of using the FrNTT, other fractional NTT could be employed. The parameters of the scheme can also be adjusted. This includes the dimension of the transform matrix, which must coincide with the dimension of the image blocks one desires to process, the field in which the transform is defined, the number of columns and rows shared by adjacent image blocks, the key length, the

[2] At this time, a comparison between the complexity of the proposed approach and that of the method given in [19] appears to be the most adequate one, because both schemes employ equivalent mathematical operations (arithmetic modulo a prime number) and similar encryption/decryption structures. Comparing the complexity of the proposed approach with those of encryption schemes based on real-valued mathematical tools, for instance, could not produce realistic results and would require considering details related to implementation, speed, memory, etc.

number of encryption rounds, etc. Such a flexibility allows designing schemes with distinct robustness levels and whose implementations require different computational efforts.

7 Concluding Remarks

A new approach for defining fractional number-theoretic transforms was presented in this paper. The method is based on matrix functions and, differently from previously proposed definitions, it does not require the construction of NTT eigenvectors. This allows us to express fractional powers of an NTT matrix as a linear combination of its integer powers. As we have demonstrated, this also enables the use of standard fast algorithms in the computation of FrNTT matrix-vector products. Our approach was developed for Fourier, Hartley, cosine and sine number-theoretic transforms, and some peculiarities of these transforms were discussed.

Besides addressing theoretic aspects, we revisited a recently proposed image encryption scheme based on the FrFNT. We have shown how the FrFNT constructed by means of matrix functions can be employed in such an application and emphasized the advantages that it provides. After carrying out computer experiments and performing a preliminary security analysis, we have concluded that the scheme remains resistant against the main cryptographic attacks.

Currently, we are investigating further theoretic and practical aspects concerning fractional number-theoretic transforms. The possibility of defining FrNTT based on closed-form Hermite–Gaussian eigenvectors over finite fields [16] and the characterization, in the number-theoretic scenario, of fractional convolution and other important properties [41] have been studied. Applications of FrNTT in the fields of error-correcting codes, cryptography, digital watermarking, and multiuser communication have also been investigated [10,13,23,24,36].

Acknowledgements This research was partially supported by Conselho Nacional de Desenvolvimento Científico e Tecnológico, under Grants 456744/2014-2 and 307686/2014-0.

8 Appendix

8.1 Proof of Lemma 1

Since the integer powers of $\mathbf{S}$ are

$$\begin{aligned}\mathbf{S}^2 &= \left[\frac{1}{2}(1-j)\mathbf{I} + \frac{1}{2}(1+j)\mathbf{P}\right]^2 = \mathbf{P},\\ \mathbf{S}^3 &= \mathbf{PS}\end{aligned}$$

and

$$\mathbf{S}^4 = \mathbf{PS}^2 = \mathbf{I},$$

the matrix $\mathbf{S}$ has period 4 and, therefore, the lemma holds.

8.2 Proof of Lemma 2

Permuting the second and the last rows of $\mathbf{B}$, the updated first and second rows have zero entries at the same positions. Applying analogous permutations to other rows of $\mathbf{B}$, after an even number of permutations, one obtains the matrix

$$\mathbf{B}_p = \begin{bmatrix} b\; 0 \ldots 0\; 0 \ldots 0\; c \\ c\; 0 \ldots 0\; 0 \ldots 0\; b \\ 0\; b \ldots 0\; 0 \ldots c\; 0 \\ 0\; c \ldots 0\; 0 \ldots b\; 0 \\ \vdots\; \vdots \;\; \vdots \;\; \vdots\; \vdots \;\; \vdots \;\; \vdots\; 0 \\ 0\; 0 \ldots b\; c \ldots 0\; 0 \\ 0\; 0 \ldots c\; b \ldots 0\; 0 \end{bmatrix}, \tag{30}$$

such that $|\mathbf{B}_p| = |\mathbf{B}|$. This determinant can be expressed as $|\mathbf{B}| = \left(b^2 - c^2\right)|\mathbf{B}_s|$, where $\mathbf{B}_s$ is the matrix obtained excluding the first two rows and the first and the last columns of $\mathbf{B}_p$. We can also write $|\mathbf{B}| = \left(b^2 - c^2\right)^2 |\mathbf{B}_{s-1}|$, where $\mathbf{B}_{s-1}$ is a matrix obtained excluding the first two rows and the first and the last columns of $\mathbf{B}_s$. In general, we observe that $|\mathbf{B}| = \left(b^2 - c^2\right)^{s_0} |\mathbf{B}_{s-s_0+1}|$, $s_0 = 1, \ldots, \frac{N}{2}$, where $\mathbf{B}_{s-s_0+1}$ is a matrix obtained excluding the first two rows and the first and the last columns of $\mathbf{B}_{s-s_0+2}$. This leads us to $|\mathbf{B}| = \left(b^2 - c^2\right)^{\frac{N}{2}}$.

8.3 Proof of Lemma 3

The minimal polynomial of $\mathbf{S}$ is given by $|\lambda\mathbf{I}_N - \mathbf{S}|$. Defining $b' = \lambda - 1$, $b = \lambda - \frac{1-j}{2}$ and $c = \frac{1+j}{2}$, after permuting the rows in a way similar to that used in the proof of Lemma 2, the matrix $\lambda\mathbf{I}_N - \mathbf{S}$ has a structure similar to that of $\mathbf{B}_p$. If N is odd, the minimal polynomial of $\mathbf{S}$ is given by

$$P_{\mathbf{S}}(\lambda) = \begin{vmatrix} b'\; 0 \ldots 0\; 0 \ldots 0 \\ 0\;\; b \ldots 0\; 0 \ldots c \\ 0\;\; c \ldots 0\; 0 \ldots b \\ \vdots\;\; \vdots \;\; \vdots \;\; \vdots\; \vdots \;\; \vdots \;\; \vdots \\ 0\; 0 \ldots b\; c \ldots 0 \\ 0\; 0 \ldots c\; b \ldots 0 \end{vmatrix}. \tag{31}$$

Therefore,

$$P_{\mathbf{S}}(\lambda) = b'\left(b^2 - c^2\right)^{\frac{N-1}{2}} = (\lambda - 1)\left[(2\lambda - 1 + j)^2 - (1 + j)^2\right]^{\frac{N-1}{2}}.$$

If N is even, the minimal polynomial of $\mathbf{S}$ is given by

$$P_{\mathbf{S}}(\lambda) = \begin{vmatrix} b' & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & b' & 0 & \dots & 0 \\ 0 & b & \dots & 0 & 0 & 0 & \dots & c \\ 0 & c & \dots & 0 & 0 & 0 & \dots & b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & 0 \\ 0 & 0 & \dots & b & 0 & c & \dots & 0 \\ 0 & 0 & \dots & c & 0 & b & \dots & 0 \end{vmatrix}. \tag{32}$$

Therefore,

$$P_{\mathbf{S}}(\lambda) = (b')^2 \left(b^2 - c^2\right)^{\frac{N-2}{2}} = (\lambda - 1)^2 \left[(2\lambda - 1 + j)^2 - (1 + j)^2\right]^{\frac{N-2}{2}}.$$

8.4 Proof of Theorem 2

If N is odd, the eigenvalues of $\mathbf{S}$ are the roots of the polynomial

$$P_{\mathbf{S}}(\lambda) = (\lambda - 1) \left[(2\lambda - 1 + j)^2 - (1 + j)^2\right]^{\frac{N-1}{2}}.$$

From $(2\lambda - 1 + j)^2 - (1 + j)^2 = 0$, the roots are $\{1, -j\}$, both with multiplicity $\frac{N-1}{2}$. The root $\lambda = 1$ also occurs due to the term $(\lambda - 1)$. If N is even, the eigenvalues are the roots of the polynomial

$$P_{\mathbf{S}}(\lambda) = (\lambda - 1)^2 \left[(2\lambda - 1 + j)^2 - (1 + j)^2\right]^{\frac{N-2}{2}}.$$

From $(2\lambda - 1 + j)^2 - (1 + j)^2 = 0$, the roots are $\{1, -j\}$, both with multiplicity $\frac{N-2}{2}$. The root $\lambda = 1$ occurs twice more, due to the term $(\lambda - 1)^2$.

References

1. A. Akhshani, S. Behnia, A. Akhavan, H. Abu Hassan, Z. Hassan, A novel scheme for image encryption based on 2D piecewise chaotic maps. Opt. Commun. **283**(17), 3259–3266 (2010)
2. L.B. Almeida, The fractional Fourier transform and time-frequency representations. IEEE Trans. Signal Process. **42**(11), 3084–3091 (1994)
3. G. Bhatnagar, Q.M.J. Wu, B. Raman, A new fractional random wavelet transform for fingerprint security. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. **42**(1), 262–275 (2012)
4. D.T. Birtwistle, The eigenstructure of the number theoretic transforms. Sig. Process. **4**(4), 287–294 (1982)
5. R.E. Blahut, *Fast Algorithms for Signal Processing* (Cambridge University Press, Cambridge, 2010)
6. D.M. Burton, *Elementary Number Theory*, 7th edn. (McGraw-Hill Science/Engineering/Math, New York, 2010)
7. R.M. Campello de Souza, H.M. de Oliveira, A. Kauffman, A.J.A. Paschoal, Trigonometry in finite fields and a new Hartley transform, in *Proceedings of IEEE International Symposium on Information Theory* (ISIT'98) (IEEE, 1998), p. 293

8. C. Candan, M. Alper Kutay, H.M. Ozaktas, The discrete fractional Fourier transform. IEEE Trans. Signal Process. **48**(5), 1329–1337 (2000)
9. S.C. Chan, K.L. Ho, Direct methods for computing discrete sinusoidal transforms. IEE Proc. F Radar Signal Process. **137**(6), 433–442 (1990)
10. R.J. Cintra, V.S. Dimitrov, R.M. Campello de Souza, H.M. de Oliveira, Fragile watermarking using finite field trigonometrical transforms. Signal Process. Image Commun. **24**, 587–597 (2009)
11. E. Condon, Immersion of the Fourier transform in a continuous group of functional transformations. Proc. Natl. Acad. Sci. **23**, 158–164 (1937)
12. H.M. de Oliveira, R.M. Campello de Souza, *Coding, Communications and Broadcasting, vol. 1, chap. Orthogonal Multilevel Spreading Sequence Design* (Research Studies Press/Wiley, Taunton, 2000)
13. I. Djurović, S. Stanković, I. Pitas, Digital watermarking in the fractional Fourier transformation domain. J. Netw. Comput. Appl. **24**, 167–173 (2001)
14. N.J. Higham, *Functions of Matrices: Theory and Computation* (Society for Industrial and Applied Mathematics, Philadelphia, 2008)
15. C.W. Kok, Fast algorithm for computing discrete cosine transform. IEEE Trans. Signal Process. **45**(3), 757–760 (1997)
16. A. Kuznetsov, Explicit Hermite-type eigenvectors of the discrete Fourier transform. SIAM J. Matrix Anal. Appl. **36**(4), 1443–1464 (2015)
17. R. Lidl, H. Niederreiter, *Finite Fields. Encyclopedia of Mathematics and its Applications*, 2nd edn. (Cambridge University Press, Cambridge, 2008)
18. J.B. Lima, E.A.O. Lima, F. Madeiro, Image encryption based on the finite field cosine transform. Sig. Process. Image Commun. **28**(10), 1537–1547 (2013)
19. J.B. Lima, L.F.G. Novaes, Image encryption based on the fractional Fourier transform over finite fields. Sig. Process. **94**(1), 521–530 (2014)
20. J.B. Lima, R.M. Campello de Souza, Finite field trigonometric transforms. Appl. Algebra Eng. Commun. Comput. **22**(5–6), 393–411 (2011)
21. J.B. Lima, R.M. Campello de Souza, The fractional Fourier transform over finite fields. Sig. Process. **92**(2), 465–476 (2012)
22. J.B. Lima, R.M. Campello de Souza, Fractional cosine and sine transforms over finite fields. Linear Algebra Appl. **438**(8), 3217–3230 (2013)
23. J.B. Lima, R.M. Campello de Souza, D.C. Cunha, Multiuser communication based on the discrete fractional Fourier transform, in *Proceedings of IEEE International Conference on Communications* (ICC'2012) (Ottawa, Canada 2012), pp. 3569–3573
24. J.B. Lima, R.M. Campello de Souza, D. Panario, The eigenstructure of finite field trigonometric transforms. Linear Algebra Appl. **435**(8), 1956–1971 (2011)
25. J.B. Lima, R.M.C. de Souza, P.H.E.S. Lima, Fractional number-theoretic transform based on matrix functions, in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP'2014) (Florence, Italy, 2014), pp. 2614–2618
26. S.A. Martucci, Symmetric convolution and the discrete sine and cosine transforms. IEEE Trans. Signal Process. **42**(5), 1038–1051 (1994)
27. D. Mendlovic, H.M. Ozaktas, Fractional Fourier transforms and their optical implementation: I. J. Opt. Soc. Am. A **10**, 1875–1881 (1993)
28. V. Namias, The fractional order Fourier transform and its application in quantum mechanics. J. Inst. Math. Appl. **25**, 241–265 (1980)
29. H.M. Ozaktas, Z. Zalevsky, M. Alper Kutay, *The Fractional Fourier transform: with Applications in Optics and Signal Processing* (Wiley, London, 2001)
30. S.C. Pei, J.J. Ding, Fractional cosine, sine, and Hartley transforms. IEEE Trans. Signal Process. **50**(7), 1661–1680 (2002)
31. S.C. Pei, C.C. Tseng, M.H. Yeh, J.J. Shyu, Discrete fractional Hartley and Fourier transforms. IEEE Trans. Circuits Syst. II Analog Digital Signal Process. **45**(6), 665–675 (1998)
32. S.C. Pei, C.C. Wen, J.J. Ding, Closed-form orthogonal eigenvectors generated by complete generalized Legendre sequences. IEEE Trans. Circuits Syst. I Regul. Pap. **55**(11), 3469–3479 (2008)
33. S.C. Pei, C.C. Wen, J.J. Ding, Closed form orthogonal number theoretic transform eigenvectors and the fast fractional NTT. IEEE Trans. Signal Process. **59**(5), 2124–2135 (2011)
34. S.C. Pei, M.H. Yeh, The discrete fractional cosine and sine transforms. IEEE Trans. Signal Process. **49**(6), 1198–1207 (2001)
35. J.M. Pollard, The fast Fourier transform in a finite field. Math. Comput. **114**(25), 82–100 (1971)

36. N. Rutter, S. Boussakta, A. Bystrov, Assessment of the one-dimensional generalized new Mersenne number transform for security systems, in *Proceedings of IEEE 77th Vehicular Technology Conference* (VTCSpring) (Dresden, Germany, 2013), pp. 1–5
37. B. Santhanam, J.H. McClellan, The discrete rotational Fourier transform. IEEE Trans. Signal Process. **44**(4), 994–998 (1996)
38. N. Smart, *ECRYPT II yearly report on algorithms and keysizes (2011–2012)*. Technical Report (European Network of Excellence in Cryptology II, 2012)
39. N. Wiener, Hermitian polynomials and Fourier analysis. J. Math. Phys. **8**, 70–73 (1929)
40. G. Ye, Image scrambling encryption algorithm of pixel bit based on chaos map. Pattern Recogn. Lett. **31**(5), 347–354 (2010)
41. A.I. Zayed, A convolution and product theorem for the fractional Fourier transform. IEEE Signal Process. Lett. **5**(4), 101–103 (1998)