



ELSEVIER

Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

SCIENCE @ DIRECT®

Microprocessors and Microsystems 27 (2003) 511–521

MICROPROCESSORS AND  
MICROSYSTEMS

[www.elsevier.com/locate/micpro](http://www.elsevier.com/locate/micpro)

# The fractional Fourier transform: theory, implementation and error analysis

V. Ashok Narayanan<sup>a,1</sup>, K.M.M. Prabhu<sup>b,\*</sup>

<sup>a</sup>Department of Electrical and Computer Engineering, University of Maryland at College Park, College Park, MD 20742, USA

<sup>b</sup>Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai 600036, India

Received 6 August 2002; revised 15 May 2003; accepted 4 June 2003

## Abstract

The fractional Fourier transform is a time–frequency distribution and an extension of the classical Fourier transform. There are several known applications of the fractional Fourier transform in the areas of signal processing, especially in signal restoration and noise removal. This paper provides an introduction to the fractional Fourier transform and its applications. These applications demand the implementation of the discrete fractional Fourier transform on a digital signal processor (DSP). The details of the implementation of the discrete fractional Fourier transform on ADSP-2192 are provided. The effect of finite register length on implementation of discrete fractional Fourier transform matrix is discussed in some detail. This is followed by the details of the implementation and a theoretical model for the fixed-point errors involved in the implementation of this algorithm. It is hoped that this implementation and fixed-point error analysis will lead to a better understanding of the issues involved in finite register length implementation of the discrete fractional Fourier transform and will help the signal processing community make better use of the transform.

© 2003 Elsevier B.V. All rights reserved.

*Keywords:* Fourier transform; Discrete fractional Fourier transform; Implementation; ADSP-2192 processor; Error analysis

## 1. Introduction

The purpose of this paper is to provide a brief introduction to the fractional Fourier transform (FRFT) and then follow it up with its discrete version and give the details of its discrete implementation on ADSP-2192. An accurate theoretical fixed-point error analysis of the discrete implementation is also provided. The FRFT is a generalization of the ordinary Fourier transform [1–3] with an order parameter  $\alpha$  and is identical to the ordinary Fourier transform when this order  $\alpha$  is equal to  $\pi/2$ . Since the ordinary Fourier transform and related techniques are of importance in various different areas like communications, signal processing and control systems, it is natural to expect the FRFT to find many applications in these fields as well. In fact, the FRFT has already found many applications in the areas of signal processing and communications [2–4]. This

paper will provide a general overview of the various properties of the FRFT and then go on to describe the discrete fractional Fourier transform (DFRFT). This paper provides a glimpse of the various applications of the FRFT, which make its implementation on a digital signal processor (DSP) worthwhile. The paper also lists the details of the implementation of the DFRFT on an ADSP-2192 processor and then provides an accurate model for the fixed-point error analysis.

## 2. The fractional Fourier transform

The FRFT belongs to the class of time–frequency representations that have been extensively used by the signal processing community. In all the time–frequency representations, one normally uses a plane with two orthogonal axes corresponding to time and frequency. If we consider a signal  $x(t)$  to be represented along the time axis and its ordinary Fourier transform  $X(f)$  to be represented along the frequency axis, then the Fourier transform operator (denoted by  $F$ ) can be visualized as a change in representation of the signal corresponding to a

\* Corresponding author. Tel.: +91-44-2257-9368; fax: +91-44-2257-0509.

E-mail addresses: [prabhu\\_kmm@hotmail.com](mailto:prabhu_kmm@hotmail.com) (K.M.M. Prabhu), [ascii\\_ashok@hotmail.com](mailto:ascii_ashok@hotmail.com) (V. Ashok Narayanan).

<sup>1</sup> He was with the Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai 600036, India.

counterclockwise rotation of the axis by an angle  $\pi/2$ . This is consistent with some of the observed properties of the Fourier transform. For example, two successive rotations of the signal through  $\pi/2$  will result in an inversion of the time axis. Moreover, four successive rotations will leave the signal unaltered since a rotation through  $2\pi$  of the signal should leave the signal unaltered. The FRFT is a linear operator that corresponds to the rotation of the signal through an angle which is not a multiple of  $\pi/2$ , i.e. it is the representation of the signal along the axis  $u$  making an angle  $\alpha$  with the time axis. The detailed development of the FRFT is established in Refs. [1–3].

2.1. Definition

The FRFT is defined with the help of the transformation kernel  $K_\alpha$ , as

$$K_\alpha(t, u) = \begin{cases} \delta(t - u) & \text{if } \alpha \text{ is a multiple of } 2\pi \\ \delta(t + u) & \text{if } \alpha + \pi \text{ is a multiple of } 2\pi \\ \sqrt{\frac{1 - j \cot \alpha}{2\pi}} e^{j((u^2 + t^2)/2) \cot \alpha - j ut \operatorname{cosec}(\alpha)} & \text{if } \alpha \text{ is not a multiple of } \pi \end{cases} \quad (1)$$

Another useful form of writing the square root factor preceding the transformation kernel  $K_\alpha$  can be obtained using the relation

$$\sqrt{\frac{1 - j \cot \alpha}{2\pi}} = \sqrt{\frac{-j e^{j\alpha}}{2\pi \sin \alpha}} \quad (2)$$

The FRFT is defined using this kernel as (FRFT of order  $\alpha$  of  $x(t)$  denoted by  $X_\alpha(u)$ )

$$X_\alpha(u) = \int_{-\infty}^{\infty} x(t) K_\alpha(t, u) dt, \quad (3)$$

where

$$X_\alpha(u) = \begin{cases} \sqrt{\frac{1 - j \cot \alpha}{2\pi}} e^{j(u^2 \cot \alpha/2)} \int_{-\infty}^{\infty} x(t) e^{j(t^2 \cot \alpha/2) - j ut \operatorname{cosec}(\alpha)} dt & \text{if } \alpha \text{ is not a multiple of } \pi \\ x(t) & \text{if } \alpha \text{ is a multiple of } 2\pi \\ x(-t) & \text{if } \alpha + \pi \text{ is a multiple of } 2\pi \end{cases} \quad (4)$$

The definition is discussed in detail in Refs. [1,2].

2.2. Computation of the fractional Fourier transform

The FRFT of a signal  $x(t)$  as given by Eq. (4) can be computed by the following steps:

1. product by a chirp—chirps are functions whose frequency is linearly increasing with time
2. a Fourier transform with its argument scaled by  $\operatorname{cosec}(\alpha)$
3. another product by a chirp
4. a product by a complex constant

The problem of the existence of the FRFT has been widely studied [2] and it is found that the FRFT of a signal  $x(t)$  exists under the same conditions in which its Fourier transform exists. In order to discuss the various properties of the FRFT, it would be ideal to denote the FRFT in an operator notation.

2.3. Properties of the fractional Fourier transform

Let  $F_\alpha$  denote the operator corresponding to the FRFT of angle  $\alpha$ . Under this notation, some of the important properties of the FRFT operator are listed below:

- (a) *Identity operator.*  $F_0$  is the identity operator. The FRFT of order  $\alpha = 0$  is the input signal itself. The FRFT of order  $\alpha = 2\pi$  corresponds to the successive application of the ordinary Fourier transform 4 times and therefore also acts as the identity operator, i.e.  $F_0 = F_{\pi/2} = I$ .
- (b) *Fourier transform operator.*  $F_{\pi/2}$  is the Fourier transform operator. The FRFT of order  $\alpha = \pi/2$  gives the Fourier transform of the input signal.
- (c) *Successive applications of FRFT.* Successive applications of FRFT are equivalent to a single transform whose order is equal to the sum of the individual orders.

$$F_\alpha(F_\beta) = F_{\alpha+\beta}.$$

- (d) *Inverse.* The FRFT of order  $-\alpha$  is the inverse of the FRFT of order  $\alpha$  since  $F_{-\alpha}(F_\alpha) = F_{\alpha-\alpha} = F_0 = I$ .

Property (a) follows from the direct definition of the kernel for  $\alpha = 0$ . Property (b) can be proved by expanding

the Kernel for the special case when  $\alpha = \pi/2$ . Property (c) can be proved using the convolution property of the kernel. Property (d) is a consequence of properties (b), (c) and the fact that four consecutive applications of the Fourier transform correspond to the Identity operator. Property (d) can also be proved using property (c).

One important conclusion that can be drawn from these properties is that the signal  $x(t)$  and its FRFT (of order  $\alpha$ )  $X_\alpha(u)$  form a transform pair and are related to each other by the following equation:

$$X_\alpha(u) = \int_{-\infty}^{\infty} x(t)K_\alpha(u, t)dt, \tag{5}$$

$$x(t) = \int_{-\infty}^{\infty} X_\alpha(u)K_{-\alpha}(u, t)du. \tag{6}$$

These properties are discussed in detail in Refs. [1,2].

#### 2.4. Fractional Fourier transform of related signals

For the discussion that follows, let us assume that  $x(t)$  and  $X_\alpha(u)$  form a FRFT pair. The FRFTs of signals related to  $x(t)$  are listed in Table 1.

#### 2.5. Parseval's theorem

The well-known Parseval's theorem for Fourier transform can be easily extended to the FRFT as well.

Table 1  
Fractional Fourier transforms of related signals

| Operation                | Signal, $x(t)$      | Fractional Fourier transform, $X_\alpha(u)$   |
|--------------------------|---------------------|---|
| Time shift               | $x(t - \tau)$       | $e^{j(\tau^2/2)\sin \alpha \cos \alpha - j u \tau \sin \alpha} X_\alpha(u - \tau \cos \alpha)$  |
| Modulation               | $x(t)e^{j vt}$      | $e^{-j v^2 (\sin \alpha \cos \alpha)/2 + j uv \cos \alpha} X_\alpha(u - v \sin \alpha)$   |
| Multiplication with ramp | $tx(t)$             | $u \cos \alpha X_\alpha(u) + j \sin \alpha X'_\alpha(u)$  |
| Inversion of time axis   | $x(-t)$             | $X_\alpha(-u)$  |
| Scaling of time axis     | $x(ct)$             | $\sqrt{\frac{1 - j \cot \alpha}{c^2 - j \cot \alpha}} e^{j(u^2/2)\cot \alpha(1 - (\cos^2 \beta/\cos^2 \alpha))} X_\beta\left(\frac{u \sin \beta}{c \sin \alpha}\right)$<br>where $\cot \beta = \frac{\cot \alpha}{c^2}$ |
| Differentiation          | $x'(t)$             | $X'_\alpha(u)\cos \alpha + j u \sin \alpha X_\alpha(u)$   |
| Integration              | $\int_a^t x(t')dt'$ | $\sec \alpha e^{-j(u^2/2)\tan \alpha} \int_a^u X_\alpha(z)e^{j(z^2/2)\tan \alpha} dz$<br>if $\alpha - \pi/2$ is not a multiple of $\pi$   |
| Division by ramp         | $x(t)/t$            | $-j \sec \alpha e^{j(u^2/2)\cot \alpha} \int_{-\infty}^u x(z)e^{-j(z^2/2)\cot \alpha} dz$<br>if $\alpha$ is not a multiple of $\pi$   |

$$\int_{-\infty}^{\infty} x(t)y^*(t)dt = \int_{-\infty}^{\infty} X_\alpha(u)Y_\alpha^*(u)du. \tag{7}$$

An interesting observation that develops from the Parseval's theorem is the energy preservation property as given below. This property can be obtained by the application of Parseval's theorem

$$\int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty} |X_\alpha(u)|^2 du. \tag{8}$$

The energy preservation property of the FRFT is only to be expected because the FRFT is based on the decomposition of the signal on the orthonormal basis set of the chirp functions. Due to the energy preserving property of the Fourier transform, the squared magnitude of the Fourier transform of a signal is often called the energy spectrum of the signal and is interpreted as the distribution of the signal energy among the different frequencies. Although less intuitive, this allows us to call the squared magnitude of the FRFT of a signal as the fractional energy spectrum of the signal and interpret it as the distribution of the signal energy between the different chirp basis functions.

#### 2.6. Real signals

The symmetry properties of the Fourier transform for even and odd real sequences do not extend to the FRFT, since the FRFT of a real function is not necessarily Hermitian. The symmetry property for the FRFT is slightly different and is shown below.

Let  $x(t)$  be real.

$$\begin{aligned} X_\alpha^*(u) &= \left( \sqrt{\frac{1 - j \cot \alpha}{2\pi}} \right)^* \int_{-\infty}^{\infty} x^*(t) e^{-j((u^2+t^2)/2)\cot \alpha + j ut \operatorname{cosec}(\alpha)} dt \\ &= \sqrt{\frac{1 - j \cot(-\alpha)}{2\pi}} \int_{-\infty}^{\infty} x(t) e^{j((u^2+t^2)/2)\cot(-\alpha) - j ut \operatorname{cosec}(-\alpha)} dt \\ &= X_{-\alpha}(u). \end{aligned} \tag{9}$$

### 3. Some simple FRFTs

#### 3.1. Gaussian

It is a well-documented fact that the FRFT of a signal corresponds to the rotation of the Wigner distribution of that signal by the required angle  $\alpha$  [1]. Moreover, it is also known that the Wigner distribution of the Gaussian is a two-dimensional Gaussian in the time–frequency plane. Therefore, any rotation of this Gaussian by any angle  $\alpha$  will leave the signal unaltered. Thus, it is intuitive to expect that the FRFT of a Gaussian is a Gaussian. The FRFT of a Gaussian of various orders can be seen in Fig. 1.

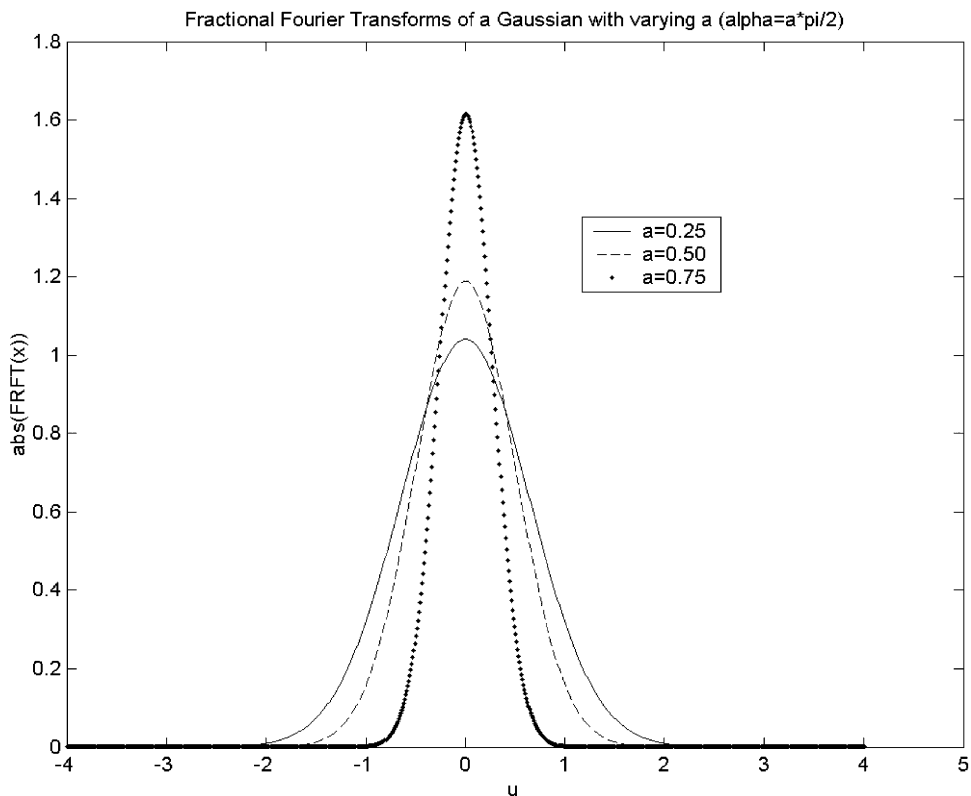


Fig. 1. Fractional Fourier transforms of a Gaussian with  $\alpha = \pi/8$ ,  $\alpha = \pi/4$  and  $\alpha = 3\pi/8$ .

3.2. Rectangular signal

We know that the Fourier transform of a rectangular signal is a sinc function. We also know that the FRFT of any signal is a continuous function of the parameter  $\alpha$ . Therefore, it would be interesting to observe how the FRFT of the rectangular signal evolves into a sinc function as  $\alpha$  is increased from 0 to  $\pi/2$ . This evolution can be observed from Fig. 2.

3.3. Other common signals

The analytical expressions for the FRFTs of various other common signals are calculated and tabulated in Table 2.

4. Applications

The FRFT, being just an extension of the classical Fourier transform, can be used effectively in all situations where the Fourier transform is presently being used. Some gains can be expected in most of these applications because of the additional degree of freedom (angle of rotation) that the FRFT provides us with. In this paper, we will discuss some of typical applications of the FRFT in

the area of signal processing that have been developed over time.

4.1. Swept frequency system

The swept frequency filters are commonly used, for example, in frequency analyzers for high-frequency signals. Swept frequency filters are linear time-varying systems that can be represented in the form shown in Fig. 3. They can also be represented by their time-varying impulse response  $h(t, \tau)$ , which is the response at time  $t$  to an input  $\delta(t, \tau)$ .

$$h(t, \tau) = e^{j(c/2)(t^2 - \tau^2)} g(t - \tau),$$

$$y(t) = \int_{-\infty}^{\infty} x(\tau)h(t, \tau)d\tau.$$

Let  $\alpha = -\cot^{-1}c$  and  $G(u)$  the Fourier transform of  $g(t)$

$$\begin{aligned} Y_{\alpha}(u) &= \sqrt{\frac{1 - j \cot \alpha}{2\pi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(\tau)h(t, \tau) \\ &\quad \times e^{j((u^2 + \tau^2)/2)\cot \alpha - j u \tau \operatorname{cosec}(\alpha)} d\tau dt \\ &= \sqrt{\frac{1 - j \cot \alpha}{2\pi}} \int_{-\infty}^{\infty} x(\tau)e^{j((u^2 + \tau^2)/2)\cot \alpha} \\ &\quad \times G(u \operatorname{cosec}(\alpha))e^{-j u \tau \operatorname{cosec}(\alpha)} d\tau \\ &= G(u \operatorname{cosec}(\alpha))X_{\alpha}(u). \end{aligned} \tag{10}$$

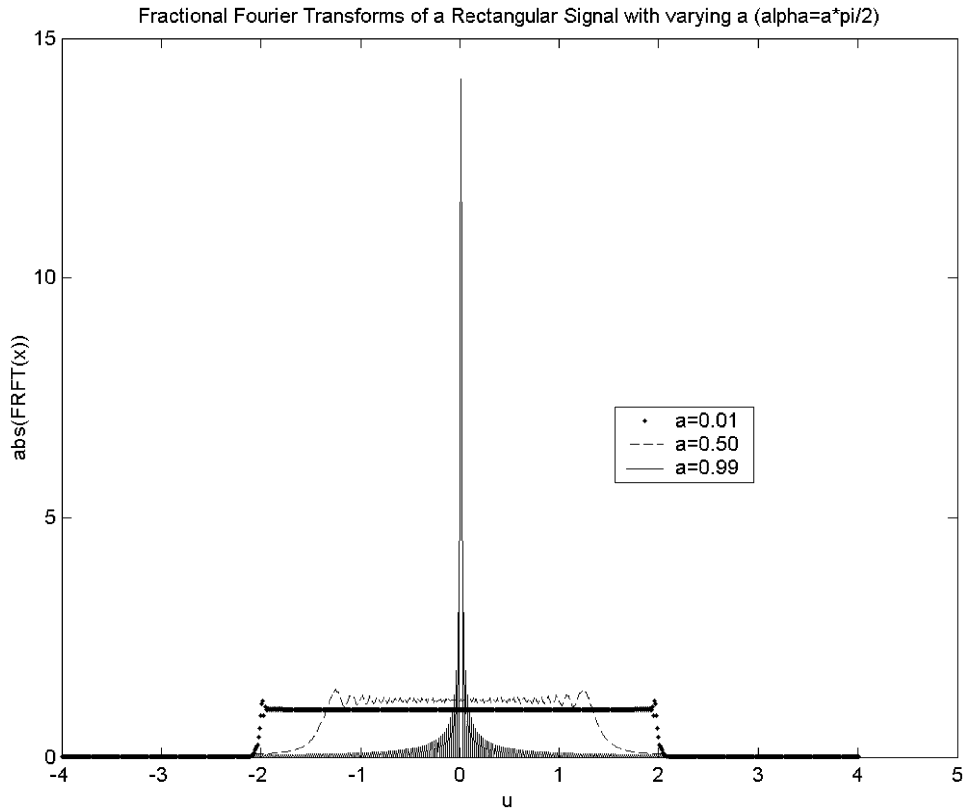


Fig. 2. Fractional Fourier transforms of a rectangular signal ( $\alpha = 0.005\pi$ ,  $\alpha = \pi/4$  and  $\alpha = 0.99 * \pi/2$ ).

Therefore,  $G(u \operatorname{cosec}(\alpha))$  can be called the transfer function of the swept frequency filter in the fractional Fourier domain. This observation allows us to treat the time-variant transfer function of the swept frequency filters in much the same way as we treat time-invariant filters using the classical Fourier transform.

#### 4.2. Optimal filtering

A very elegant and important application of the FRFT has been in optimal filtering [4]. This improves the performance of the optimal Wiener filter especially in the case where the corrupting filter is time-variant. The results

Table 2  
Fractional Fourier transforms of some simple signals

| Signal   | Fractional Fourier transform with angle $\alpha$  | Condition   |
|--|---|---|
| 1 $\delta(t - \tau)$                             | $\sqrt{\frac{1 - j \cot \alpha}{2\pi}} e^{j((\tau^2 + u^2)/2) \cot \alpha - j u \tau \operatorname{cosec} \alpha}$  | If $\alpha$ is not a multiple of $\pi$                      |
| 2 1  | $\sqrt{1 + j \tan \alpha} e^{-j(u^2/2) \tan \alpha}$  | If $\alpha - \pi/2$ is not a multiple of $\pi$              |
| 3 $e^{j u \tau}$                                 | $\sqrt{1 + j \tan \alpha} e^{-j((v^2 + u^2)/2) \tan \alpha + j u v \sec \alpha}$  | If $\alpha - \pi/2$ is not a multiple of $\pi$              |
| 4 $e^{j c(t^2/2)}$                               | $\sqrt{\frac{1 + j \tan \alpha}{1 + c \tan \alpha}} e^{j(u^2/2)(c - \tan \alpha)/(1 + c \tan \alpha)}$  | If $\alpha - \arctan(c) - \pi/2$ is not a multiple of $\pi$ |
| 5 $e^{-(t^2/2)}$                                 | $e^{-(u^2/2)}$  |   |
| 6 $H_n(t)e^{-(t^2/2)}$ $H_n$ —Hermite polynomial | $e^{-j n \alpha} H_n(u) e^{-(u^2/2)}$   |   |
| 7 $e^{-c(t^2/2)}$                                | $\sqrt{\frac{1 - j \cot \alpha}{c - j \cot \alpha}} e^{j(u^2/2)(c^2 - 1) \cot \alpha / (c^2 + \cot^2 \alpha)} e^{-(u^2/2)(c \operatorname{cosec}^2(\alpha) / (c^2 + \cot^2 \alpha))}$ |   |

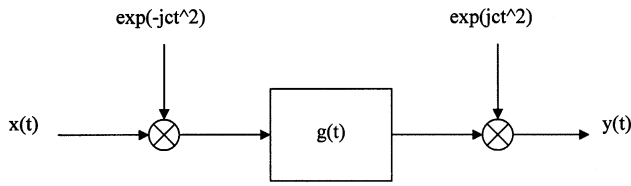


Fig. 3. Swept frequency system.

$$\frac{-1}{4\pi} \begin{pmatrix} -2 & 1 & 0 & \dots & 0 & 1 \\ 1 & 2 \cos(2\pi/n) - 4 & 1 & \dots & 0 & 0 \\ 0 & 1 & 2 \cos(\frac{2\pi}{n} 2) - 4 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 0 & 0 & \dots & 1 & 2 \cos(\frac{2\pi}{n}(n-1)) - 4 \end{pmatrix}. \tag{11}$$

showing the better performance of a Wiener filter using FRFT is provided along with the analytical framework for the same, in Ref. [4].

4.3. Other applications

A discrete version of the fractional sine and cosine transforms has been proposed in Ref. [5] and it is therefore expected that significant progress will be made in the area of image processing. Some interesting applications of FRFT in the areas of control systems and telecommunications are now being probed into. It is, therefore, quite reasonable to expect that the FRFT will eventually replace Fourier transform as a single most potent tool in signal analysis. Thus it becomes exceedingly important both to develop and to implement the discrete version of the FRFT.

5. The discrete fractional Fourier transform

In the recent years, the FRFT has gained great significance as a tool for signal processing [4–8]. Therefore, many attempts have been made to find the discrete version of the FRFT [6,7]. Though quite a few different algorithms were suggested by various researchers, the most consistent definition that agrees with the various properties of the FRFT and approximates to the FRFT is the one proposed by Candan et al. in Ref. [6].

This definition of the DFRFT has the following properties:

- (a) Unitarity
- (b) Index additivity
- (c) Reduction to DFT when  $\alpha = \pi/2$

In this paper, we just state the analytical method to calculate the DFRFT. The theoretical framework and the justification for the proposed algorithm is provided in Ref. [6].

5.1. The algorithm for computing the DFRFT

- (a) Generate the **S** matrix

The **S** matrix is defined as below:

$$\begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{11}$$

- (b) Generate the **P** matrix

The **P** matrix is defined below for **P** odd and even. This matrix decomposes an arbitrary vector  $f(n)$  into its even and odd components. The **P** matrix, maps the even part of the  $N$ -dimensional vector to the first  $\lfloor (N/2 + 1) \rfloor$  components and the odd part to the remaining components. For example, the **P** matrix of dimension 5 is given by

$$\mathbf{P} = \frac{1}{\sqrt{2}} \begin{pmatrix} \sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}. \tag{12}$$

- (c) Calculate the **Ev** and **Od** matrix defined by

$$PSP^{-1} = PSP = \begin{pmatrix} Ev & 0 \\ 0 & Od \end{pmatrix}. \tag{13}$$

Once the **Ev** and **Od** matrices are calculated as above, their eigenvectors and eigenvalues are computed. Sort the eigenvectors of the **Ev** and **Od** matrices in the descending order of eigenvalues and denote the sorted eigenvectors as  $E_k$  and  $O_k$ , respectively.

- (d) Calculate the contents of the **U** matrix using the equations given below:

$$Let U_{2k}[n] = P[e_k^T \mathbf{0}]^T, \tag{14}$$

where  $\mathbf{0}$  is a zero vector of length  $N - \lfloor (N/2 + 1) \rfloor$  to make  $U$  a  $N \times N$  matrix.

$$Let U_{2k+1}[n] = P[0 \dots 0 o_k^T]^T.$$

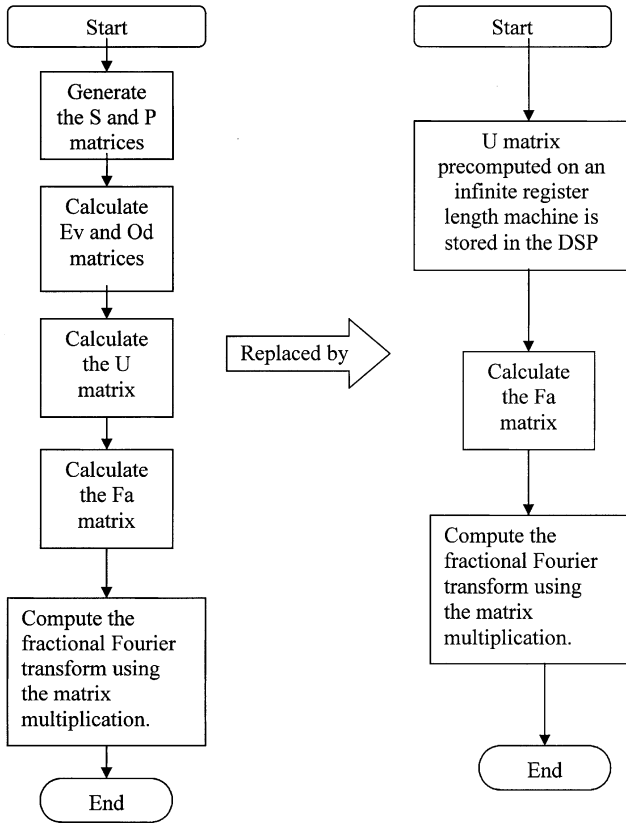


Fig. 4. Flow charts of the original and refined method to compute DFRFT.

(e) The FRFT operator matrix is defined as

$$F^a[m, n] = \sum_{k \in \mu} U_k[m] e^{-j(\pi/2)ka} U_k[n], \quad (15)$$

$$\mu = \{0, 1, 2, \dots, N - 2, (N - (N)_2)\}.$$

(f) The FRFT of order  $a$  is then calculated by a matrix multiplication

$$f_a = F^a f. \quad (16)$$

The flow chart for the implementation of the DFRFT is shown in Fig. 4. An  $O(N \log N)$  algorithm for the digital computation of FRFT is given in Ref. [7], which computes the samples of the continuous FRFT and hence is strictly not the discrete FRFT (DFRFT).

Table 3  
Properties of DFRFT

|   |                             |                         |                               |
|---|-----------------------------|-------------------------|-------------------------------|
| 0 | $f[n]$                      | $\leftrightarrow^a$     | $f_a[n]$                      |
| 1 | $f[n] + g[n]$               | $\leftrightarrow^a$     | $f_a[n] + g_a[n]$             |
| 2 | $f_a[n]$                    | $\leftrightarrow^b$     | $f_{a+b}[n]$                  |
| 3 | $f[n]$                      | $\leftrightarrow^{a=1}$ | DFT $\{f[n]\}$                |
| 4 | $f[-n]$                     | $\leftrightarrow^a$     | $f_a[-n]$                     |
| 5 | $f^*[n]$                    | $\leftrightarrow^a$     | $f_a^*[n]$                    |
| 6 | Even $\{f[n]\}$             | $\leftrightarrow^a$     | Even $\{f_a[n]\}$             |
| 7 | Odd $\{f[n]\}$              | $\leftrightarrow^a$     | Odd $\{f_a[n]\}$              |
| 8 | $\sum_{n=0}^{N-1}  f[n] ^2$ | =                       | $\sum_{n=0}^{N-1}  f_a[n] ^2$ |

## 5.2. Properties of the discrete fractional Fourier transform

In Table 3, we list a number of additional useful properties of the DFRFT, which are extensions of the corresponding properties of the discrete Fourier transform.

## 6. Implementation of the DFRFT

The DFRFT has been implemented on ADSP-2192 using the ADSP-2192 Instruction set and the inbuilt C pre-processor provided by the DSP. The ADSP-2192 is a fixed-point DSP and this introduces fixed-point computation errors on the DSP. Moreover, it also provides the results of simulation of the fixed-point errors that have been observed due to the limitation on the number of bits used for each computation. The implementation is extremely important for applications that have been discussed in Refs. [1–4,8].

### 6.1. Method to minimize the computations and fixed-point limitations

Since the ADSP-2192 is a fixed-point DSP, there is a limitation on the number of bits available for storing multiplication and addition, and this results in some loss in accuracy. We will now discuss a method that minimizes both the loss in accuracy and the number of computations involved. The algorithm for the computation of the  $F_a$  matrix has been described in the previous section. In implementing this algorithm, fixed-point limitations introduce errors at every step of the algorithm. In the algorithm for calculating the  $N$ -point FRFT matrix of order  $a$ , the intermediate matrix  $U$  is independent of the order of the FRFT  $a$ . This observation can be used to reduce the errors involved in the computation.

In most practical applications, the DSP is used to repeatedly calculate an  $N$ -point DFT or DFRFT of a given sequence where the length of DFRFT ( $N$ ) is a fixed number. For example, in many typical applications the continuous stream of data is windowed using a  $N$ -point rectangular (or any other similar) window and this windowing operation is followed by the DFRFT of the windowed signal. Therefore, it is sufficient for all practical purposes to be able to calculate the FRFT of various orders  $a$  or  $\alpha$ , for a given length  $N$ . This length  $N$  is fixed depending upon the application in question. This can be efficiently realized by precomputing the  $U$  matrix for a given length  $N$  and storing the  $U$  matrix on the DSP. This method of storing the  $U$  matrix instead of computing it saves computations and also minimizes the fixed-point errors obtained.

The calculation of the  $U$  matrix involves the computation of the eigenvectors and eigenvalues of two matrices of order  $[(N/2 + 1)]$  and  $[(N/2 - 1)]$ , respectively. The calculation of the eigenvectors is very computationally intensive. There are no known algorithms of order less than or equal to  $O(N^2)$  for the computation of eigenvalues and eigenvectors.

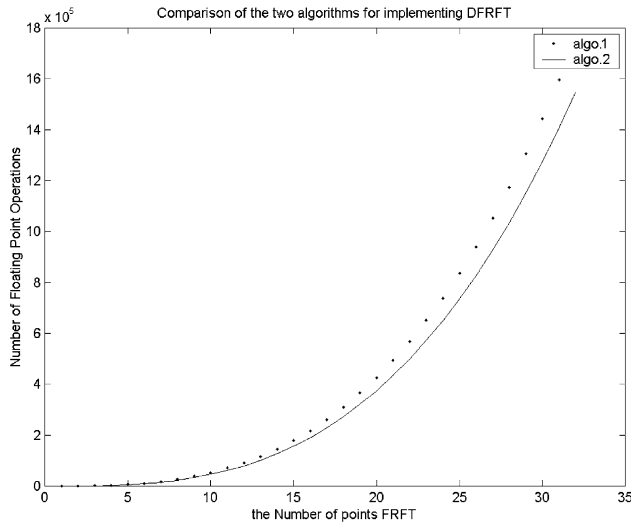


Fig. 5. Comparison of computational complexity of the two algorithms.

Therefore, this calculation takes a major part of the total computations involved in the algorithm for computing the DFRFT matrix. By storing the actual  $\mathbf{U}$  matrix instead of computing it, a significant amount of computations are reduced. Moreover, this also means that the calculation of the  $\mathbf{U}$  matrix need not be a real-time calculation. Therefore, this can be done on an infinite length (or a very high word-length) machine so that the values of the  $\mathbf{U}$  matrix are available with very high precision. Thus, this method also helps in containing the errors involved due to the fixed-point implementation of the DFRFT on ADSP-2192 processor. But this method is useful only when the sample length remains constant.

In this modified method, the eigenvalues and the eigenvectors of the various matrices are computed only once and stored in memory instead of being computed every time a DFRFT is needed. This results in a significant reduction in the number of additions and multiplications that have to be performed for calculating the DFRFT. Thus we gain speed at the expense of memory. In general, the complexity of computations of an algorithm is measured by the number of floating point operations required for the algorithm. From Fig. 5 the tremendous saving in the number of floating point operations due to this particular implementation is evident. This also provides an idea about the memory requirements of this algorithm when compared to that of the original.

## 7. Fixed-point error analysis

An electronic digital computer or a DSP processes information in binary format. When the scientific programming convention is used in a fixed-point machine, the binary number is stored in the 2's complement form, where the width of quantization  $q$  and the word-length ( $M$ ) of

the computer are related as  $q = 2^{-(M-1)}$ . For fixed-point arithmetic in a signal processing context, it is natural to consider a register as representing a fixed-point fraction. In this way, the product of two numbers remains a fraction and truncating or rounding the least significant bits can maintain the limited register length. The result of addition on fixed-point fractions need not be truncated or rounded but it can increase in magnitude so that the sum eventually is not a fraction. This effect is commonly referred to as overflow, and can be handled by restricting the input data to be sufficiently small so that the possibility of overflow is avoided. Due to the finite word-length constraints, the performance of the DSP is inevitably degenerated and the mathematical formulation of this degeneration is discussed in Refs. [9–11]. First, when data are read into the DSP, a quantization error is incurred. Second, a round-off error arises in the evaluation of every product in the computation. As a consequence, the FRFT obtained varies from that which would have been obtained with an infinite word-length DSP. For the purpose of comparison, it is convenient to define the 'ideal' or the 'error-less' FRFT as a realization by an infinite word-length machine.

### 7.1. Theoretical analysis of fixed-point error

The eigenvectors and the eigenvalues of the  $\mathbf{E}_k$  and  $\mathbf{O}_k$  matrices are calculated using an infinite word-length machine and stored in the DSP as described earlier. These eigenvectors and eigenvalues are rounded to  $m (= b + 1)$  bits when they are stored in the DSP. This rounding leads to an error in the values stored in the DSP. Let us assume that the error in this rounding off operation is uniformly distributed between  $-q/2$  and  $q/2$ , where  $q = 2^{-b} = 2^{-(m-1)}$ . This means that the error in each element of  $\mathbf{E}_k$  or  $\mathbf{O}_k$  matrix is uniformly distributed between  $-2^{-m}$  and  $2^{-m}$ .

Therefore

$$\sigma_{E_k}^2 = \frac{q^2}{12} = \frac{2^{-2b}}{12}. \quad (17)$$

As can be seen from the algorithm for computing the DFRFT, each element of the  $\mathbf{U}$  matrix is a weighted sum of two elements of the  $\mathbf{E}_k$  or the  $\mathbf{O}_k$  matrices, the weighting factor being  $1/\sqrt{2}$ .

$$U_{2k}[n] = P[e_k^T \mathbf{0} \cdot \cdot]^T, \quad (18)$$

$$U_{2k+1}[n] = P[\mathbf{0} \cdot \cdot \mathbf{0} o_k^T]^T,$$

$$\text{i.e. } U[l, m] = \frac{1}{\sqrt{2}} [E_{(i,m)} + E_{(j,m)}],$$

where  $i$  and  $j$  are the two non-zero elements of  $l$ th row of  $P$ . Therefore

$$\sigma_U^2 = \frac{1}{2} [\sigma_{E_k}^2 + \sigma_{E_k}^2], \quad (19)$$

$$\sigma_U^2 = \sigma_{E_k}^2 = \frac{q^2}{12} = \frac{2^{-2b}}{12}.$$

Moreover, the probability distribution of the error in the  $\mathbf{U}$  matrix is triangular, since the  $\mathbf{U}$  matrix is obtained by a weighted sum of two uniform distributions. This is because when two random distributions are added the probability distribution of their sum corresponds to the convolution of their individual probability distributions which in the present case is a uniform probability distribution.

The FRFT matrix is calculated from the  $\mathbf{U}$  matrix using

$$F^a[m, n] = \sum_{k \in \mu} U_k[m] e^{-j(\pi/2)ka} U_k[n], \quad (20)$$

$$\mu = \{0, 1, 2, \dots, N - 2, (N - (N)_2)\}.$$

We already know the variance of the error involved in the fixed-point implementation of the DFRFT matrix. Now we need to find the variance of the fixed-point error in the DFRFT matrix. A careful observation leads us to note that this can indeed be interpreted as a convolution between two sequences of  $U_k$ 's each of which have a variance of error given by  $\sigma_{E_k}^2$ . The results for the variance of fixed-point error obtained during the convolution operation between a sequence with error was discussed in Refs. [9–11]. But in our case both sequences that are being convolved have fixed-point error involved in their representations. Therefore, applying the results shown in Refs. [9–11], we get

$$F^a[m, n] = \sum_{k \in \mu} U_k[m] e^{-j(\pi/2)ka} U_k[n], \quad (21)$$

$$\mu = \{0, 1, 2, \dots, N - 2, (N - (N)_2)\},$$

$$\Rightarrow \sigma_{F_a}^2 = 4\sigma_U^2 \sum_{i=1}^N [U_k[m]]^2.$$

The probability distribution of the error in  $\mathbf{U}$  matrix is triangular. This is so because each element of  $\mathbf{U}$  is the sum of two elements of the  $E$  matrix and the error distribution of the  $E$  matrix is uniform.

$$\text{But } \sum_{i=1}^N [U_k[m]]^2 = NE[U_k^2],$$

where  $E[x]$  is the mean of  $x$

$$\begin{aligned} E[U_k^2] &= \int_{-\sqrt{2}}^{\sqrt{2}} p(U_k) U_k^2 dU_k \\ &= 2 \int_0^{\sqrt{2}} U_k^2 \left[ \frac{1}{\sqrt{2}} - \frac{U_k}{2} \right] dU_k = \frac{1}{3}. \end{aligned} \quad (22)$$

Therefore

$$\sigma_{F_a}^2 = 4\sigma_{U_k}^2 \frac{N}{3} = \frac{4N}{3} \left( \frac{q^2}{12} \right) = \frac{4N}{9} (2^{-2m}). \quad (23)$$

### 7.2. Simulation

The noise introduced in the calculation of the DFRFT can be calculated either directly or by simulation. In the direct

method, the DFRFT matrix is calculated on a finite word-length DSP and the 'ideal' DFRFT matrix is calculated on an infinite word-length machine. The difference between these two values gives the error involved due to the finite register length. A simulation of the finite word-length effect can also be used. In this case both the 'ideal' and the 'actual' DFRFT matrices are calculated on the same infinite word-length machine. In the calculation of the 'actual' DFRFT matrix, all the computations are done as they would be on a finite word-length DSP, i.e. the result of each multiplication is restricted to  $m$ -bits where  $m$  is the word-length of the machine. The effect of overflow is also incorporated in the simulations. If an overflow occurs at any computation then the result of the corresponding register is set to 1.

The simulation results that are presented in this paper have been obtained by following the second method. The second order statistics of the noise involved in the computation of the DFRFT matrix were studied. The error, i.e. the difference between the 'ideal' and the actual

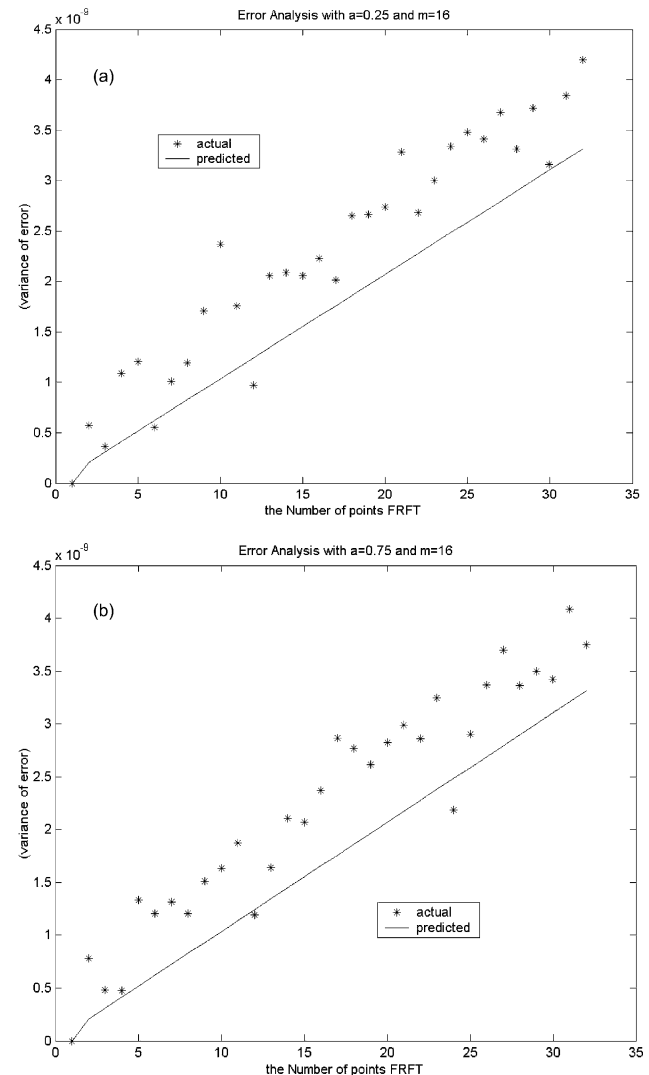


Fig. 6. Variance of the finite register length error in computation of DFRFT with  $N$ . (a)  $m = 16$  and  $a = 0.25$ , (b)  $m = 16$  and  $a = 0.75$ .

DFRFT matrix is calculated. Both the predicted and the actual variance of the error is plotted.

The variance of the error is dependent both on the number of points of the DFRFT and also the number of registers for storing each number. As can be seen from the equation derived, the variance of the error increases linearly with  $N$ . The variance of the error also decreases exponentially with the  $m$ , the word-length of the DSP.

The variance of the noise in the computation of the DFRFT matrix is plotted in Figs. 6 and 7. Both the predicted and the actual variances are plotted and the two of them are almost identical thus proving that the theoretical estimate was indeed accurate. The  $x$ -axis is  $N$ , the length of the DFRFT that is being computed. Very similar results are obtained for different values of  $m$ , the register length. For instructional purposes, this variation is shown for register lengths  $m = 16$  and  $32$ .

The figures given (Figs. 6a,b and 7a,b) plot the variance of noise in the fixed-point implementation as a function of the register length for a given transform length,  $N$ . Since

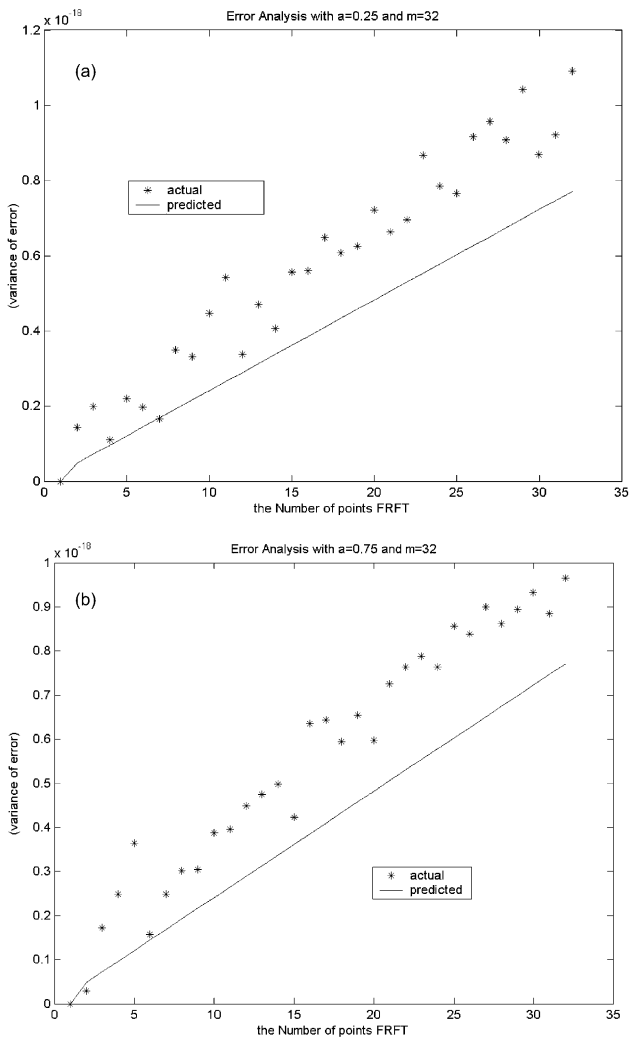


Fig. 7. Variance of the finite register length error in computation of DFRFT with  $N$ . (a)  $m = 32$  and  $a = 0.25$ , (b)  $m = 32$  and  $a = 0.75$ .

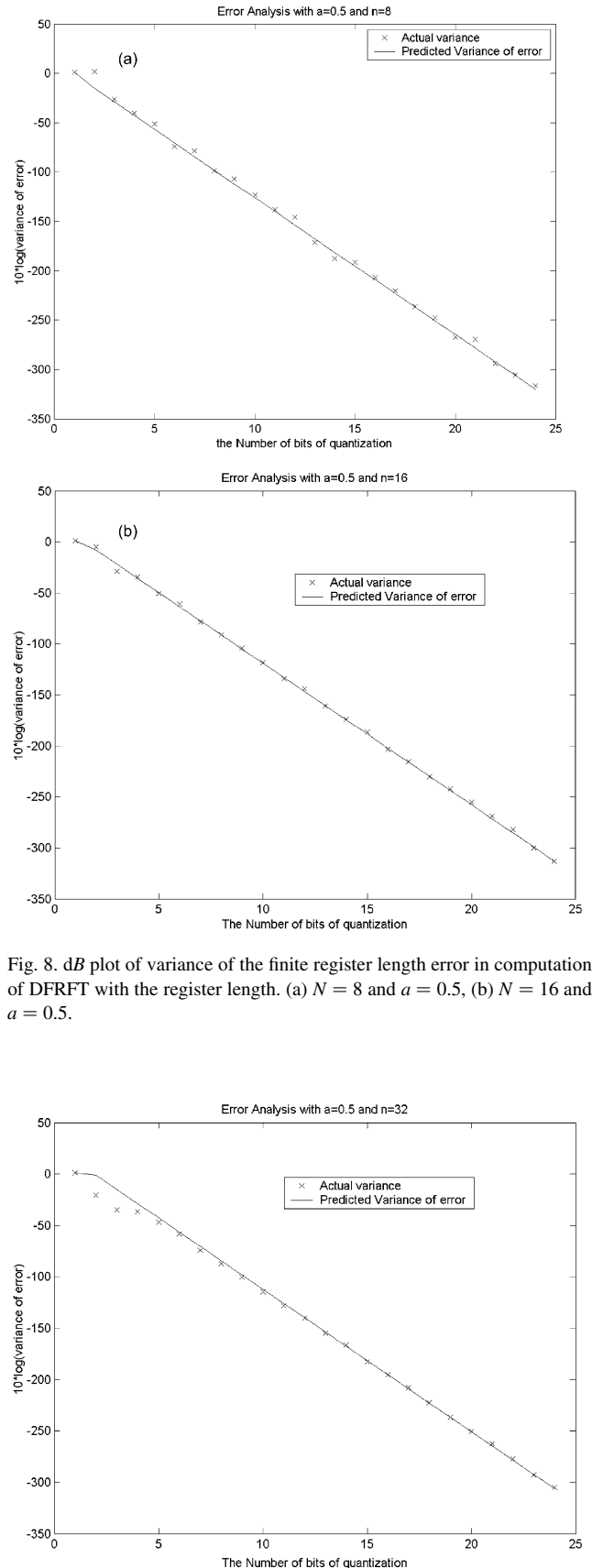


Fig. 8. dB plot of variance of the finite register length error in computation of DFRFT with the register length. (a)  $N = 8$  and  $a = 0.5$ , (b)  $N = 16$  and  $a = 0.5$ .

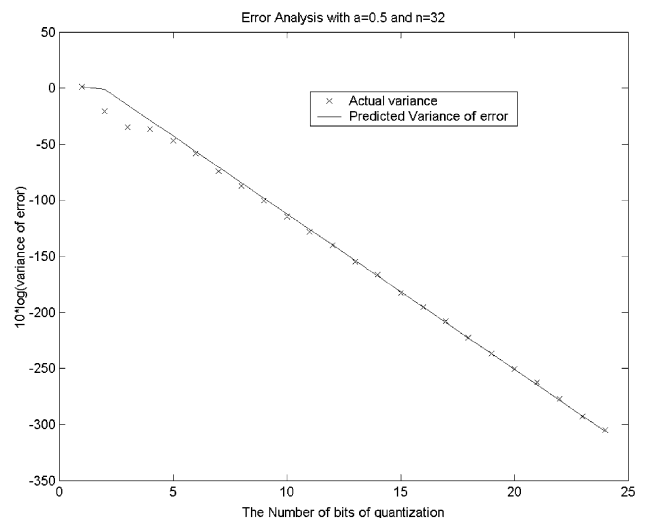


Fig. 9. dB plot of variance of the finite register length error in computation of DFRFT with the register length ( $N = 32$  and  $a = 0.5$ ).

the variance of the error decreases exponentially with the register length, it is much more informative to study the dB plot of this variance. In Figs. 8a,b and 9, the variance of noise in decibel scale ( $10 \log_{10}(\text{variance of noise})$ ) is plotted as a function of the register length with the transform length  $N$  kept constant. The results are shown for  $N = 8, 16$  and  $32$  with  $a = 0.5$  for instructional purposes. Similar results are obtained for all the other values of  $a$  also. In this case also it is seen that the theoretical prediction is fairly accurate.

## 8. Conclusion

This paper presents a brief introduction to the theory and applications of the FRFT and then provides the details of one particular implementation of the DFRFT on ADSP-2192. The noise in this implementation due to the finite register length of the ADSP-2192 has also been studied and a theoretical model for the second order statistics of this noise have been developed. It is hoped that this implementation of the DFRFT will help the signal processing community make better use of the FRFT.

## References

- [1] L.B. Almeida, The fractional Fourier transform and time–frequency representations, *IEEE Trans. Signal Proc.* 42 (11) (1994) 3084–3093.
- [2] H.M. Ozaktas, Z. Zalevsky, M.A. Kutay, *The Fractional Fourier Transform with Applications in Optics and Signal Processing*, Wiley, New York, 2001.
- [3] D. Mendlovic, *Advances in Imaging and Electron Physics*, vol. 106, Academic Press, New York, 1999.
- [4] M.A. Kutay, H.M. Ozaktas, O. Arikan, L. Onural, Optimal filtering in fractional Fourier domains, *IEEE Trans. Signal Proc.* 45 (5) (1997) 1129–1143.
- [5] S.C. Pei, M.H. Yeh, The discrete fractional cosine and sine transforms, *IEEE Trans. Signal Proc.* 49 (6) (2001) 1198–1207.
- [6] C. Candan, M.A. Kutay, H.M. Ozaktas, The discrete fractional Fourier transform, *IEEE Trans. Signal Proc.* 48 (5) (2000) 1329–1337.
- [7] H.M. Ozaktas, O. Arikan, M.A. Kutay, G. Bozdagi, Digital computation of the fractional Fourier transform, *IEEE Trans. Signal Proc.* 44 (9) (1996) 2141–2149.
- [8] M.F. Erden, M.A. Kutay, H.M. Ozaktas, Repeated filtering in consecutive fractional Fourier domains and its application to signal restoration, *IEEE Trans. Signal Proc.* 47 (5) (1999) 1458–1462.
- [9] B. Liu, Effect of finite word length on accuracy of digital filters: a review, *IEEE Trans. Circuit Theory* 18 (6) (1971) 670–677.
- [10] A.V. Oppenheim, C.J. Weinstein, Effects of finite register-length in digital filtering and the fast Fourier transform, *Proc. IEEE* 60 (8) (1972) 957–976.
- [11] J.B. Knowles, E.M. Olcayto, Coefficient accuracy and digital filter response, *IEEE Trans. Circuit Theory* 15 (1) (1968) 31–41.



**V. Ashok Narayanan** received the Bachelor of Technology degree in Electrical Engineering from the Indian Institute of Technology Madras, India, in the year 2002 and he is currently pursuing his Masters degree in Electrical and Computer Engineering at the University of Maryland, College Park, USA. He is also working with the Center For Automation Research (CFAR) in the areas of Image Processing and Computer Vision.



**K.M.M. Prabhu** obtained the BSc, (Engg) degree in Electronics and Communication Engineering, the MSc (Engg) degree in Applied Electronics Engineering and the PhD degree in Digital Signal Processing (DSP), in 1971, 1973 and 1981, respectively. He joined the Department of Electrical Engineering, Indian Institute of Technology (IITM) Madras, Chennai, India, in 1981, where he is currently a full Professor. While at IIT Madras, he has been a Principal Investigator for many projects funded by the Defence Research and Development Organization (DRDO). He has published more than 95 papers in the International Journals and in the International Conference Proceedings. Under his supervision, 17 MS (by research) and PhD students have completed their degrees. He has conducted 10 short-term courses in the area of DSP for the benefit of Engineers and Scientists from various organizations as well as the Students from various Engineering Colleges. He has been a reviewer for a number of IEEE, IEE, IETE and other International Journals. His current research interests include DSP Algorithms and their Implementations, Digital Filter Structures, Time–Frequency Signal Analysis and Wavelet-Based Signal Processing. Prof. Prabhu is a Senior Member of the IEEE since 1993.