FOURIER TRANSFORM – SIGNAL PROCESSING

Edited by **Salih Mohammed Salih**

INTECHOPEN.COM

Fourier Transform – Signal Processing
Edited by Salih Mohammed Salih

Published by InTech
Janeza Trdine 9, 51000 Rijeka, Croatia

Publishing Process Manager Vana Persen
Technical Editor Teodora Smiljanic
Cover Designer InTech Design Team

First published April, 2012
Printed in Croatia

A free online edition of this book is available at www.intechopen.com
Additional hard copies can be obtained from orders@intechopen.com

Fourier Transform – Signal Processing, Edited by Salih Mohammed Salih
p. cm.
ISBN 978-953-51-0453-7

Contents

Preface

This book focuses on the Fourier transform applications in signal processing techniques. The field of signal processing has seen explosive growth during the past decades, as phenomenal advances both in research and application have been made. During the preparation of this book, we found that almost all the textbooks on signal processing have a section devoted to the Fourier transform theory. Most of those describe some formulas and algorithms, but one can easily be lost in seemingly incomprehensible mathematics. The basic idea behind all those horrible looking formulas is rather simple, even fascinating: *it is possible to form any function as a summation of a series of sine and cosine terms of increasing frequency*. In other words, any space or time varying data can be transformed into a different domain called the *frequency space*. A fellow called *Joseph Fourier* first came up with the idea in the 19th century, and it was proven to be useful in various applications, mainly in signal processing. As far as we can tell, Gauss was the first to propose the techniques that we now call the Fast Fourier Transform (FFT) for calculating the coefficients in a trigonometric expansion of an asteroid's orbit in 1805. However, it was the seminal paper by Cooley and Tukey in 1965 that caught the attention of the science and engineering community and, in a way, founded the discipline of Digital Signal Processing (DSP). While the Discrete Fourier Transform (DFT) transform can be applied to any complex valued series, in practice for large series it can take considerable time to compute, the time taken being proportional to the square of the number on points in the series. It is hard to overemphasis the importance of the DFT, convolution, and fast algorithms. The FFT may be the most important numerical algorithm in science, engineering, and applied mathematics. New theoretical results still are appearing, advances in computers and hardware continually restate the basic questions, and new applications open new areas for research. It is hoped that this book will provide the background, references and incentive to encourage further research and results in this area as well as provide tools for practical applications. One of the attractive features of this book is the inclusion of extensive simple, but practical, examples that expose the reader to real-life signal processing problems, which has been made possible by the use of computers in solving practical design problems. The aim of this book is to expand the analysis of many topics related to Fourier transform. It includes three sections:

The first section deals with *DFT and FFT algorithms and their applications*. It consists of seven chapters. The first two chapters are related to the DFT: Interpolation of DFT, and DFT coefficients estimation, while the last five chapters are related to the FFT applications: FFT-based efficient algorithms for time delay estimation, channel identification for OFDM communication system, implementing FFT and IFFT cores for OFDM communication systems, FPGA implementation of IFFT in OFDM systems, fault diagnosis of induction motors based on FFT.

The chapters of the second section discuss some advanced methods used in Fourier transform analysis which are related to the *time, frequency, and phase estimation.*

This section consists of four chapters, the chapters are related to: estimation of the periodic signal parameters in the frequency domain, the uncertainty principles in the Fourier transform, the chirp excitation for estimation of bioimpedance spectrum, and the identification of systems via simple signals.

The third section includes the Fourier transform applications in *image processing techniques*. This section consists of four chapters, the chapters are concerns to: the invariant correlation using a nonlinear filter, the pattern recognition of digital images by one-dimensional signatures, the vectorial signatures for pattern recognition, and the multidimensional features extraction methods in frequency domain.

Finally, we would like to thank all the authors who have participated in this book for their valuable contribution. Also we would like to thank all the reviewers for their valuable notes. While there is no doubt that this book may have omitted some significant findings in the Fourier transform field, we hope the information included will be useful for Electrical Engineers, Communication Engineers, Signal Processing Engineers and the Mathematicians, in addition to the academic researchers working in this field.

Salih Mohammed Salih
College of Engineering
University of Anbar
Iraq

Part 1

DFT and FFT Applications

1

Interpolation Algorithms of DFT for Parameters Estimation of Sinusoidal and Damped Sinusoidal Signals

Krzysztof Duda
AGH University of Science and Technology,
Department of Measurement and Instrumentation, Krakow,
Poland

1. Introduction

Discrete Fourier Transform (DFT) is probably the most popular signal processing tool. Wide DFT use is partly dedicated to fast Fourier Transform (FFT) algorithms (Cooley & Tukey, 1965, Oppenheim et al., 1999, Lyons, 2004). DFT may also be efficiently computed by recursive algorithms in the window sliding by one sample (Jacobsen & Lyons, 2003, Duda, 2010). Unfortunately, DFT has two main drawbacks that deteriorate signal analysis which are (Harris, 1978, Oppenheim et al., 1999): 1) spectral leakage, and 2) sampling of the continuous spectrum of the discrete signal. Spectral leakage is reduced by proper time windows, and the frequency bins between DFT bins are computed by interpolated DFT (IpDFT) algorithms, thoroughly presented in this chapter.

2. Basic theory

2.1 Signal model

IpDFT algorithms may be derived for discrete sinusoidal or damped sinusoidal signals. Discrete sinusoidal signal is defined as

$$v_n = A\cos(\omega_0 n + \varphi),\ n = 0,1,2,...,N-1, \tag{1}$$

and discrete damped sinusoidal signal is defined as

$$v_n = A\cos(\omega_0 n + \varphi)e^{-dn},\ d \geq 0,\ n = 0,1,2,...,N-1, \tag{2}$$

where $A>0$ is signal's amplitude, $0<\omega_0<\pi$ is signal's frequency in radians or radians per sample also referred as angular frequency or pulsation, and $\omega_0=\pi$ rad corresponds to the half of the sampling rate F_s in hertz, $-\pi<\varphi\leq\pi$ is the phase angle in radians, n is the index of the sample, N is the number of samples, and d is damping factor. If discrete signals (1) and (2) result from sampling analog counterparts then

$$\omega_0 = 2\pi(F_0 / F_s), \tag{3}$$

where F_0 is the frequency of analog signal, $v(t)=A\cos(2\pi F_0 t+\varphi)$ or $v(t)=A\cos(2\pi F_0 t+\varphi)e^{dF_s t}$, in hertz, F_s is sampling frequency in hertz, and t is continuous time in seconds.

In section 3 it is shown how to estimate parameters of (1) and (2) i.e. A, ω_0, φ and d with the use of DFT. If the investigation refers to analog counterpart signal than parameters of the discrete signal should be rescaled adequately, for example $F_0=F_s\,\omega_0/(2\pi)$.

2.2 DFT analysis

Equations in this section are taken from the textbook (Oppenheim et al., 1999). Fourier transform (FT) of infinite length discrete time signal x_n is defined as

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x_n e^{-j\omega n}\,, \tag{4}$$

where n is integer sample index that goes from minus to plus infinity and ω is continuous frequency in radians (angular frequency, pulsation). Continuous spectrum $X(e^{j\omega})$ defined by (4) is periodic with the period 2π. The notation $X(e^{j\omega})$, instead of $X(\omega)$, stresses up the connection between FT and Z transform.

For finite length discrete time signal v_n containing N samples DFT is defined as

$$V_k = \sum_{n=0}^{N-1} v_n e^{-j(2\pi/N)kn}\,, k = 0,1,2,...,N-1\,. \tag{5}$$

From (5) it is seen, that by DFT the FT spectrum is computed only for frequencies $\omega_k=(2\pi/N)k$, that is DFT samples continuous spectrum of the discrete signal.

Finite length signal v_n, n=0,1,2,...,N-1 is obtained from infinite length signal x_n, n=...-2, -1,0,1,2,... by windowing, that is by multiplication with discrete signal w_n, called window, with nonzero values only on positions n=0,1,2,...,N-1

$$v_n = w_n x_n\,. \tag{6}$$

Signal models (1) and (2) were given with rectangular window defined as

$$w_n^R = \begin{cases} 1, & 0 \le n < N \\ 0, & 0 > n \ge N \end{cases}. \tag{7}$$

Discrete signal is always analyzed with the time window as only finite number of samples may be read into computer. If no other window is purposely used then the signal is analyzed with rectangular window (7).

Based on FT convolution property signal windowing (6) in the time domain results in following convolution in the frequency domain

$$V(e^{j\omega}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\Theta})W(e^{j(\omega-\Theta)})d\Theta\,. \tag{8}$$

where $X(e^{j\Theta})$ and $W(e^{j\Theta})$ are the FT spectra of the infinite length signal x_n and the time window w_n. Thus, according to (8) we observe the convolution of the signal spectrum with the window spectrum, and not the signal spectrum alone.

FT of infinite length signal $x_n=\cos(\omega_0 n+\varphi)$ is a pair of impulses at frequencies $\pm\omega_0+2\pi k$ thus the spectrum of the windowed sinusoidal signal (1) is

$$V(e^{j\omega}) = \frac{A_0}{2} e^{j\varphi} W(e^{j(\omega-\omega_0)}) + \frac{A_0}{2} e^{-j\varphi} W(e^{j(\omega+\omega_0)}). \tag{9}$$

Equation (9) is used as starting point in derivation of IpDFT algorithms. It is also used in leakage correction algorithms e.g. in (Radil et al., 2009).

According to (9) the spectrum $V(e^{j\omega})$ of the discrete, windowed, sinusoidal signal is the sum of two periodic replicas of window spectrum $W(e^{j\omega})$ shifted to the frequency $\pm\omega_0$ and rescaled by complex amplitude $(A_0/2)e^{\pm j\varphi}$.

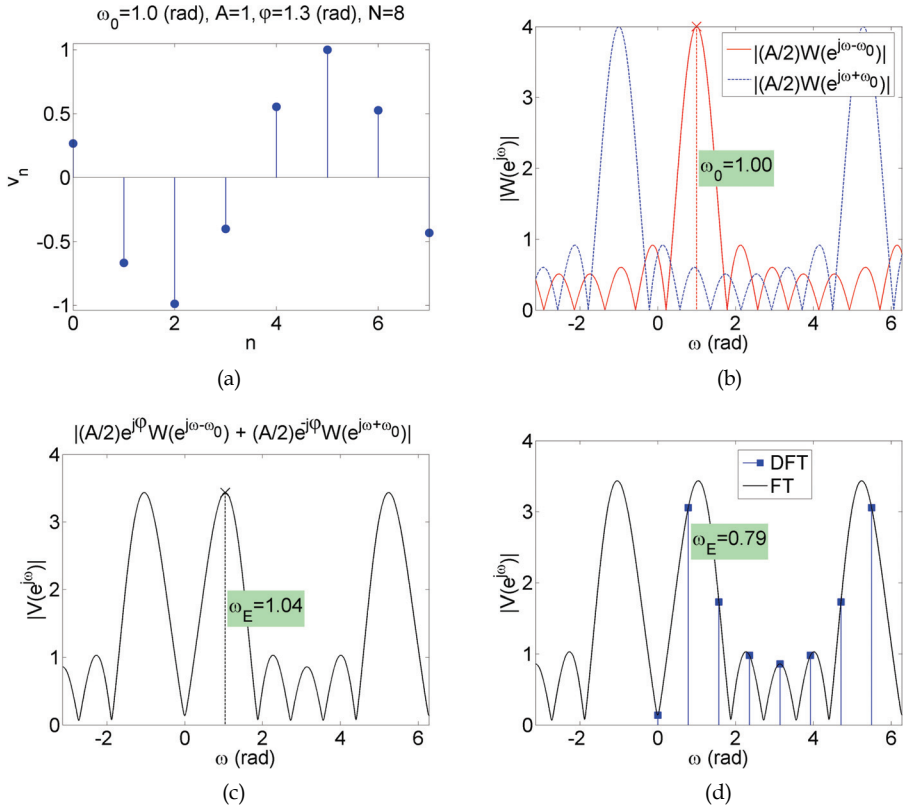


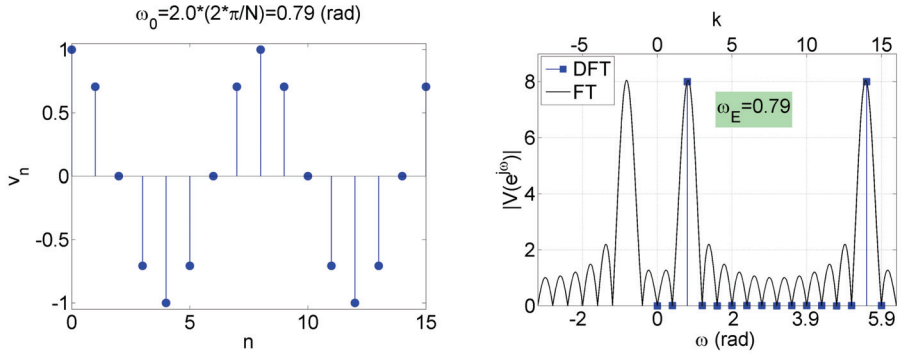
Fig. 1. Sinusoidal signal (1) with rectangular window and the modulus of its spectrum: a) signal (1), b) spectrum components for positive and negative frequencies (9), c) continuous FT spectrum (4)(9), d) sampling the continuous spectrum by DFT bins (5); ω_E denotes estimated value

Fig. 1 illustrates equations given above for sinusoidal signal (1) analyzed with rectangular window for A=1, ω_0=1 rad, φ= 1.3 rad, and N =8. Fig. 1b depicts spectrum components for positive and negative frequencies (9). The sum of those components gives continuous spectrum shown in Fig.1c that may also be computed from the FT definition (4). As seen from Fig.1c the energy is not concentrated in the single frequency bin ω_0 (as would be for infinite length observation), but spills over to all neighboring frequencies. This phenomenon is called spectral leakage and may be the reason of significant estimation errors. In the given example spectral components for positive and negative frequencies (9) influence each other and maximum shown in Fig. 1c is moved from ω_0=1 rad to ω_E=1.04 rad (E stands for estimated value), that is estimation error equals 4%. Amplitude estimation error equals approx -15%. Estimation errors depend also on signal's phase φ as the sum (9) is complex. Because of the spectral leakage, the signal in the example disturbs its own spectrum. The impact of the spectral leakage would be stronger if the signal was a sum of sinusoidal signals, as every sinusoidal component would disturb its own spectrum and the spectrum of all others sinusoidal components.

Fig. 1d depicts DFT spectrum of the signal. DFT bins are only computed for frequencies $\omega_k=(2\pi/N)k$. Frequency estimation error in that case equals approx -21%.

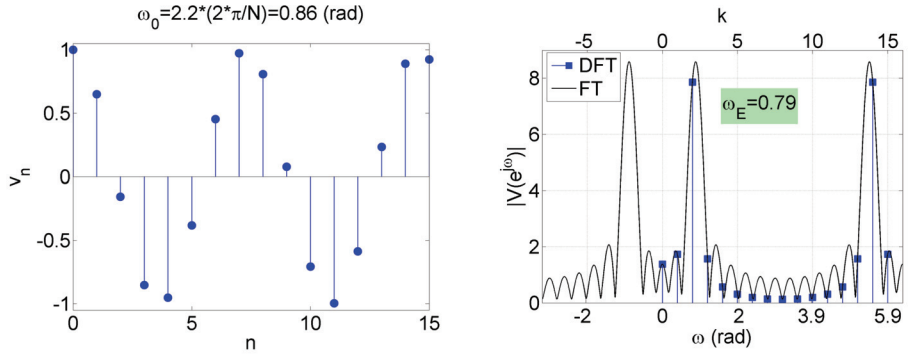
In DFT analysis spectral leakage is reduced by application of time windows other then rectangular and the error caused by the sampling of continuous spectrum only in frequencies $\omega_k=(2\pi/N)k$ is practically eliminated by IpDFT algorithms.

DFT (5) is derived for periodic signals; as a consequence frequency analysis is correct only for the signals containing integer number of periods. The signal with integer number of periods is called coherently or synchronously sampled. In field measurements, for sinusoidal signals (1) close to coherent sampling is obtained with PLL (Phase Locked Loop) that keeps integer ratio of signal's frequency F_0 to sampling frequency F_s, see (3). The frequency of coherently sampled signal is $\omega_0=(2\pi/N)k$ rad and equals the frequency of DFT bin with index k. Damped sinusoidal signals (2) have transient, and not periodic, nature and thus cannot be synchronously acquired with PLL.



(a) (b)

Fig. 2. Coherently sampled sinusoidal signal (2 periods) a), and its continuous Fourier spectrum (denoted by FT) and DFT spectrum b); ω_E - frequency of DFT bin with the highest magnitude (estimated value)



(a) (b)

Fig. 3. Non-coherently sampled sinusoidal signal a), and its continuous Fourier spectrum (denoted by FT) and DFT spectrum b); ω_E - frequency of DFT bin with the highest magnitude (estimated value)

Fig. 2a depicts coherently sampled sinusoidal signal containing exactly 2 periods. The frequency of this signal is $\omega_0=2(2\pi/N)\approx0.79$ rad and it equals the frequency of the DFT bin with index $k=2$. Fig. 2b shows continuous Fourier spectrum (4) and DFT bins of this signal. Upper OX axis is scaled in DFT index k, and lower OX axis is scaled in frequency in radians.

The range of frequencies intentionally exceeds 2π (one period) to stress up the periodic nature of the spectrum of discrete signals and the fact that spectral leakage for small frequencies originates from neighboring period (i.e. the part of the spectrum for negative frequencies). For coherent sampling of sinusoidal signal only one DFT bin is nonzero and the analysis is practically not affected by spectral leakage and sampling of the continuous spectrum.

Fig. 3a depicts sinusoidal signal which does not contain integer numbers of periods. The frequency of this signal is $\omega_0=2.2(2\pi/N)\approx0.86$ rad and lays between DFT bins with index $k=2$ and $k=3$. Fig. 3b shows FT spectrum and DFT spectrum of this signal. Estimation of signal's frequency based on the highest magnitude DFT bin is biased by the error of 0.79-0.86=-0.07 rad or -8%. High estimation errors would also be obtained for amplitude and phase estimation based on the highest magnitude DFT bin. Those errors caused by the sampling of continuous spectrum can be significantly reduced by IpDFT algorithms.

2.3 Time windows

The application of time windows in DFT analysis is thoroughly reviewed in (Harris, 1978), and also described in signal processing textbooks, e.g. (Oppenheim et al., 1999, Lyons, 2004). Other time windows with interesting properties are described in (Nuttall, 1981).

Window properties are determined by its spectrum $W(e^{j\omega})$ that consists from the main lobe which is the highest peak in the spectrum and side lobes. The shape of the spectrum of rectangular window (7) may be observed in Fig. 1b. The main lobe of the window should be as narrow as possible, and side lobes should be as low as possible. Narrow main lobe improves frequency resolution of DFT analysis, while low side lobes reduce spectral

leakage. Rectangular window is the one with the narrowest main lobe, which is an advantage and the highest side lobes which is disadvantage. All the other time windows reduce side lobes, and thus spectral leakage, by the cost of widening main lobe i.e. reducing frequency resolution. It is also known that rectangular window has the best noise immunity although systematic errors caused by leakage may be dominant for signal containing small number of cycles.

Time windows are defined as cosine windows or non cosine windows. The cosine windows may be written in the form

$$w_n = \begin{cases} \sum_{m=0}^{M} (-1)^m A_m^w \cos\left(\frac{2\pi}{N} m n\right), & 0 \le n < N \\ 0, & 0 > n \ge N \end{cases}, \tag{10}$$

where w in A_m^w is introduced to distinguish from signal's amplitude in (1-2).

IpDFT algorithms may only be analytically derived for Rife-Vincent class I (RVCI) windows. Coefficients A_m^w for RVCI windows are given in Tab.1. For M=0 RVCI window is rectangular window, and for M=1 RVCI window is Hanning (Hann) window. RVCI windows have the advantage of the fastest decay of the side lobes but they also have wide main lobe, which may be observed in Fig. 4. RVCI windows are also referred as $\cos^a(X)$, a=0,2,4,6... defined in (Harris, 1978).

m =	0	1	2	3	4	5	6
A_m^w, M=0	1						
A_m^w, M=1	1	1					
A_m^w, M=2	1	4/3	1/3				
A_m^w, M=3	1	3/2	3/5	1/10			
A_m^w, M=4	1	8/5	4/5	8/35	1/35		
A_m^w, M=5	1	105/63	60/63	45/126	5/63	1/126	
A_m^w, M=6	1	396/231	495/462	110/231	33/231	6/231	1/462

Table 1. Coefficients A_m^w for Rife-Vincent class I windows (10)

Other examples of popular cosine windows are Hamming window (M=1, A_0^w=0.54, A_1^w=0.46), and Blackman window (M=2, A_0^w=0.42, A_1^w=0.5, A_2^w=0.08).

Cosine windows (10) are the sum of frequency modulated rectangular windows, thus the spectrum of order M cosine window is

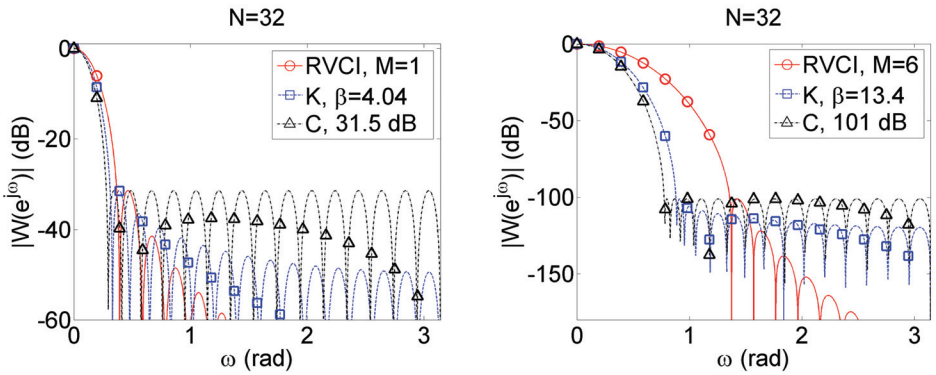
$$W_M(e^{j\omega}) = \sum_{m=0}^{M} (-1)^m \frac{A_m^w}{2} W^R(e^{j(\omega-\omega_m)}) + (-1)^m \frac{A_m^w}{2} W^R(e^{j(\omega+\omega_m)}). \tag{11}$$

where ω_m=(2π/N)m and $W^R(e^{j\omega})$ is the spectrum of rectangular window

$$W^R(e^{j\omega}) = \sum_{n=0}^{N-1} e^{-j\omega n} = e^{-j\omega(N-1)/2} \frac{\sin(\omega N/2)}{\sin(\omega/2)}. \tag{12}$$

Optimal due to main lobe width and side lobes level are non cosine Kaiser-Bessel and Dolph-Chebyshev windows (Harris, 1978). Those windows generally perform well as highlighted in (Harris, 1978 Fig. 12).

Fig. 4 shows comparison of windows spectra for RVCI, Kaiser-Bessel and Dolph-Chebyshev windows for similar attenuation of the first side lobe approx -31.5 dB (Fig. 4a) and approx -101 dB (Fig. 4b). RVCI window has the fastest decay of side lobes and the widest main lobe. Kaiser-Bessel window has narrow main lobe and slowly decay side lobes, and Dolph-Chebyshev window has the narrowest main lobe and the side lobes on the same level. The markers in Fig. 4 denote DFT bins. Cosine RVCI window has M+1 nonzero DFT coefficients as stated by (11). For non cosine Kaiser-Bessel and Dolph-Chebyshev windows all DFT coefficients are non zero.



(a) (b)

Fig. 4. Amplitude spectra of RVCI, Kaiser-Bessel (K) and Dolph-Chebyshev (Ch) windows with similar attenuation of the first side lobe, markers denote DFT bins; a) attenuation approx -31.5 dB, b) attenuation approx -101 dB

3. Interpolated DFT algorithms

IpDFT algorithm for sinusoidal signal analyzed with rectangular window was introduced in (Jain et al., 1979). In (Grandke, 1983) similar derivation was presented for sinusoidal signal analyzed with Hanning window. IpDFT algorithms for higher order RVCI windows are given in (Andria et al., 1989). In (Offelli & Petri, 1990) IpDFT algorithm for arbitrary cosine window was proposed based on polynomial approximation. In (Agrež, 2002) multipoint IpDFT was introduced with the feature of reducing long range leakage and thus reducing systematic estimation errors. IpDFT algorithm for the signal analyzed with arbitrary, even non cosine window was given in (Duda, 2011a).

IpDFT algorithms for damped sinusoidal signal analyzed with rectangular window are described in (Yoshida et al., 1981, Bertocco et al., 1994). In (Duda et al., 2011b) those algorithms were put in the same framework, and new algorithms for damped signal with rectangular window were proposed. Application of RVCI windows for damped signals in IpDFT algorithms was discussed in (Agrež, 2009, Duda et al., 2011b).

The IpDFT problem for sinusoidal and damped sinusoidal signals is depicted in Fig. 5 and may be formulated as follows. Based on the DFT spectrum V_k (5) of the signal x_n analyzed with the known window w_n (6) find the frequency correction δ so to satisfy the equation

$$\omega_0 = (k \pm \delta)\frac{2\pi}{N}, \quad 0 < \delta \le 0.5\,, \tag{13}$$

where ω_0 is signal's frequency, N is the number of samples and k is the index of DFT bin with the highest magnitude. If $|V_k{+}1| > |V_k{-}1|$, as in Fig. 5, then there is '+' in (13).

For coherent sampling depicted in Fig. 2 frequency correction δ=0, and there is no need to use IpDFT algorithm. For the example of non coherent sampling shown in Fig. 3 frequency correction is δ=0.2 and it would be δ=-0.2 if the signal's frequency was $\omega_0=1.8(2\pi/N)\approx 0.71$ rad.

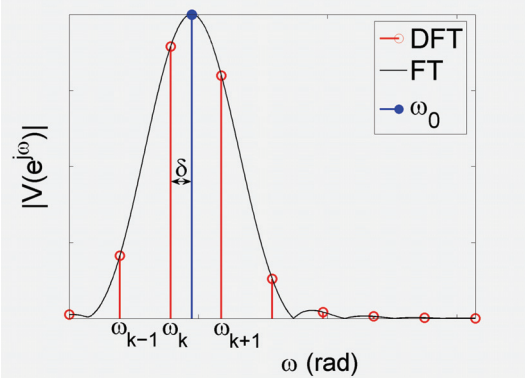


Fig. 5. Illustration of IpDFT problem; ω_{k-1}, ω_k, ω_{k+1} frequencies of DFT bins, ω_0 - signal's frequency, δ - frequency correction

Let us define the following ratio of modulus of DFT bins with the highest amplitude

$$\frac{|V_{k+1}|}{|V_k|} = \frac{|V(\omega_{k+1})|}{|V(\omega_k)|} = \frac{|V(\omega_0 - \delta 2\pi/N + 2\pi/N)|}{|V(\omega_0 - \delta 2\pi/N)|}, \tag{14}$$

where it was used $\omega_k=\omega_0-\delta 2\pi/N$, $\omega_{k+1}=\omega_0-\delta 2\pi/N+2\pi/N$ that goes from Fig. 5 and (13). As stated by (9) the spectrum of the sinusoidal signal consists from the spectrum of the window moved from the position ω=0 to $\omega=\omega_0$, thus the ratio of signal DFT (14) may be substituted by the ratio of window DFT

$$R(\delta) = \frac{|V_{k+1}|}{|V_k|} \approx \frac{|W(-\delta 2\pi/N + 2\pi/N)|}{|W(-\delta 2\pi/N)|}. \tag{15}$$

The approximation sign '≈' is used in (15) instead of equality, because the ratios may slightly differ due to spectral leakage. Solving (15) for frequency correction δ we obtain two-point (2p) IpDFT formulas. Analytic solution of (15) is only possible for RVCI windows.

The ratio of DFT bins may also be defined with three bins as

$$\frac{|V_k|+|V_{k+1}|}{|V_k|+|V_{k-1}|}=\frac{|V(\omega_0-\delta 2\pi/N)|+|V(\omega_0-\delta 2\pi/N+2\pi/N)|}{|V(\omega_0-\delta 2\pi/N)|+|V(\omega_0-\delta 2\pi/N-2\pi/N)|}, \tag{16}$$

which may be substituted by

$$R(\delta)=\frac{|V_k|+|V_{k+1}|}{|V_k|+|V_{k-1}|}\approx\frac{|W(-\delta 2\pi/N)|+|W(-\delta 2\pi/N+2\pi/N)|}{|W(-\delta 2\pi/N)|+|W(-\delta 2\pi/N-2\pi/N)|}. \tag{17}$$

Solving (17) for δ we obtain three-point (3p) IpDFT algorithm.

Let us define the following summation

$$S_{k+n}=\left|\sum_{m=-M}^{M}\frac{(-1)^{m+n}A'_m}{\delta-(m+n)}\right|,\begin{cases}\text{for } m=0\ A'_m=A_0^w\\ \text{for } m\neq 0\ A'_m=A_{|m|}^w/2\end{cases}, \tag{18}$$

where M is the order of cosine window (10) and $A_m{}^w$ is the vector of window coefficients. Then (15) and (17) may be rewritten in the form

$$\frac{|V_{k+1}|}{|V_k|}\approx\frac{S_{k+1}}{S_k} \tag{19}$$

$$\frac{|V_k|+|V_{k+1}|}{|V_k|+|V_{k-1}|}\approx\frac{S_k+S_{k+1}}{S_k+S_{k-1}} \tag{20}$$

In the next subsections IpDFT algorithms will be derived and explained. In all derivations the spectrum of the sinusoidal signal (9) is approximated by

$$V(e^{j\omega})\approx\frac{A_0}{2}e^{j\varphi}W(e^{j(\omega-\omega_0)}),\quad 0\le\omega<\pi\ . \tag{21}$$

3.1 Sinusoidal signal – RVCI windows

3.1.1 Rectangular window (RVCI *M*=0)

Rectangular window $w_n{}^R$ is defined by (7) and the spectrum of this window is given by (12). From (15) we get

$$\frac{|V_{k+1}|}{|V_k|}\approx\left|\frac{\sin(-\delta\pi+\pi)}{\sin(-\delta\pi/N+\pi/N)}\right|\left|\frac{\sin(-\delta\pi/N)}{\sin(-\delta\pi)}\right|. \tag{22}$$

By approximating sine functions by their arguments in (22) we have

$$\frac{|V_{k+1}|}{|V_k|}\approx\left|\frac{\delta\pi/N}{-\delta\pi/N+\pi/N}\right|=\left|\frac{\delta}{\delta-1}\right|. \tag{23}$$

From (23) frequency correction is

$$\delta=\frac{|V_{k+1}|}{|V_k|+|V_{k+1}|}. \tag{24}$$

Signal's frequency in next computed from (13).

The amplitude of the frequency bin ω_0 may be found from the following proportion

$$\frac{|V(\omega_0)|}{|V(\omega_k)|}=\frac{|W^R(0)|}{|W^R(\delta 2\pi/N)|}=N\left|\frac{\sin(\delta\pi/N)}{\sin(\delta\pi)}\right|\approx\left|\frac{\delta\pi}{\sin(\delta\pi)}\right|. \tag{25}$$

From (25) signal's amplitude is

$$|V(\omega_0)|=|V_k|\left|\frac{\delta\pi}{\sin(\delta\pi)}\right|=|V_k|\frac{\pi}{\sin(\delta\pi)}\delta\,. \tag{26}$$

The phase of the frequency bin ω_0 may be found from the following equation

$$\arg\{V(\omega_0)\}-\arg\{V(\omega_k)\}=\arg\{W^R(0)\}-\arg\{W^R(\delta 2\pi/N)\}=-\arg\{e^{-j\delta(\pi/N)(N-1)}\}\,. \tag{27}$$

From (27) signal's phase is

$$\arg\{V(\omega_0)\}=\arg\{V(\omega_k)\}\pm\arg\{e^{-j\delta(\pi/N)(N-1)}\}\,. \tag{28}$$

The sign '+' or '-' in (28) is selected the same way as in (13).

In similar way, for the three-point interpolation we get from (17) and (12)

$$\frac{|V_k|+|V_{k+1}|}{|V_k|+|V_{k-1}|}\approx\frac{|\delta+1|}{|\delta-1|} \tag{29}$$

and finally

$$\delta=\frac{|V_{k+1}|+|V_{k-1}|}{2|V_k|-|V_{k-1}|+|V_{k+1}|}\,. \tag{30}$$

3.1.2 Hanning (Hann, RVCI *M*=1) window

Periodic Hanning window is defined as

$$w_n^H=\begin{cases}0.5-0.5\cos\left(\dfrac{2\pi}{N}n\right), & 0\le n<N,\\ 0, & 0>n\ge N.\end{cases} \tag{31}$$

Hanning window (31) may be interpreted as the sum of rectangular window and frequency modulated rectangular window, thus based on FT properties, the spectrum of the Hanning window is the following sum of the spectra of rectangular windows

$$W^H(e^{j\omega})=-0.25W^R(e^{j(\omega-\omega_1)})+0.5W^R(e^{j\omega})-0.25W^R(e^{j(\omega+\omega_1)}),\quad \omega_1=2\pi/N \tag{32}$$

Inserting (12) into (32), taking the approximation $e^{j(\pi/N)(N-1)}\approx -1+j\,\pi/N$, and assuming $\pi/N<<1$ the modulus of Hanning window spectrum is

$$|W^H(e^{j\omega})| \approx |\sin(\omega N/2)| \left| -\frac{0.25}{\sin(\omega/2-\pi/N)} + \frac{0.5}{\sin(\omega/2)} - \frac{0.25}{\sin(\omega/2+\pi/N)} \right| \tag{33}$$

By calculating modulus of DFT bins in (15) we get

$$\frac{|V_{k+1}|}{|V_k|} = \left| \frac{0.25}{\delta} - \frac{0.5}{\delta-1} + \frac{0.25}{\delta-2} \right| \Big/ \left| -\frac{0.25}{\delta+1} + \frac{0.5}{\delta} - \frac{0.25}{\delta-1} \right|, \tag{34}$$

which may be rewritten as

$$\frac{|V_{k+1}|}{|V_k|} = \left| \frac{\delta^2(2\cdot 0.25-0.5)+\delta(2\cdot 0.5-4\cdot 0.25)+2\cdot 0.25}{\delta(\delta-1)(\delta-2)} \right| \left| \frac{\delta(\delta-1)(\delta+1)}{\delta^2(0.5-2\cdot 0.25)-0.5} \right|. \tag{35}$$

As seen in (35) coefficients by the powers of δ equal zero, and we get simple equation

$$\frac{|V_{k+1}|}{|V_k|} = \left| \frac{\delta+1}{-\delta+2} \right|, \tag{36}$$

with the solution for frequency correction

$$\delta = \frac{2|V_{k+1}|-|V_k|}{|V_k|+|V_{k+1}|}. \tag{37}$$

Signal's frequency is next computed from (13).

Similarly to (26), signal's amplitude is

$$|V(\omega_0)| = |V_k| \frac{|W^H(0)|}{|W^H(\delta 2\pi/N)|} = |V_k| \frac{0.5\pi}{\sin(\delta\pi)} \Big/ \left| -\frac{0.25}{\delta-1} + \frac{0.5}{\delta} - \frac{0.25}{\delta+1} \right|. \tag{38}$$

Signal's phase, computed as by (28), is

$$\arg\{V(\omega_0)\} = \arg\{V(\omega_k)\} \pm \arg\{e^{-j\delta(\pi/N)(N-1)}\} \tag{39}$$

with the difference that in (28) frequency correction (24) is used and in (39) frequency correction (37) is used.

3.1.3 Higher order RVCI windows

Rectangular window and Hanning window are RVCI windows of order M=0 and M=1. For rectangular window M=0, $A_0{}^w$=1 and from (18) $S_{k-1} = \frac{1}{|\delta+1|}$, $S_{k+0} = S_k = \frac{1}{|\delta|}$, $S_{k+1} = \frac{1}{|\delta-1|}$. IpDFT formulas for two-point and three-point interpolation may next be calculated from (19) and (20). For Hanning window M=1, $A_0{}^w$=0.5, $A_1{}^w$=0.5 and from (18) $S_{k+0} = S_k = \left| \frac{-0.25}{\delta+1} + \frac{0.5}{\delta} + \frac{-0.25}{\delta-1} \right|$ and $S_{k+1} = \left| \frac{0.25}{\delta} + \frac{-0.5}{\delta-1} + \frac{0.25}{\delta-2} \right|$.

For higher order cosine windows (M>1) we may write down (19) and demand that the coefficients by the powers of δ equal zero, as in (35) for Hanning window, and next solve the equations for δ. Described procedure allows us to find cosine window coefficients $A_m{}^w$ that give analytic IpDFT solutions, and it turns out that by this procedure RVCI windows are found. For example, for M=2 grouping coefficient by the powers of δ gives

$$\begin{cases} (-A_0^w + 2A_1^w - 2A_2^w)\delta^4 = 0 \\ (-4A_0^w + 8A_1^w - 8A_2^w)\delta^3 = 0 \\ (-A_0^w + 4A_1^w - 10A_2^w)\delta^2 = 0 \end{cases}, \quad \begin{cases} A_0^w = 6A_2^w \\ A_1^w = 4A_2^w \\ A_2^w \end{cases}, \tag{40}$$

which are RVC1 M=2 coefficients listed in Tab. I.

For the signal analyzed with RVCI window for two-point interpolation we get

$$\delta = \frac{(M+1)|V_{k+1}| - M|V_k|}{|V_k| + |V_{k+1}|}, \ M = 0,1,2,\ldots. \tag{41}$$

For M=0 and M=1 (41) agrees with previously derived formulas (24) and (37) for rectangular and Hanning window. Signal's frequency is next computed from (13). Signal's amplitude is

$$|V(\omega_0)| = |V_k| \frac{|W(0)|}{|W(\delta 2\pi / N)|} = \frac{2\pi}{\sin(\delta\pi)} \frac{|V_k|}{S_k} \tag{42}$$

For three-point interpolation from (20) we get

$$\delta = (M+1)\frac{|V_{k+1}| - |V_{k-1}|}{2|V_k| + |V_{k-1}| + |V_{k+1}|}, \ M = 1,2,3,\ldots. \tag{43}$$

For three-point interpolation and rectangular window frequency correction δ is computed from (30) and not (43). Equation (43) does not hold for rectangular window, i.e. M=0, because the spectrum of rectangular window contains only one nonzero DFT bin.

Signal's amplitude for three-point interpolation may be computed from the proportion

$$\begin{aligned} &\frac{|V(\omega_0 - 2\pi/N)| + 2|V(\omega_0)| + |V(\omega_0 + 2\pi/N)|}{|V_{k-1}| + 2|V_k| + |V_{k+1}|} = \\ &= \frac{|W(-2\pi/N)| + 2|W(0)| + |W(2\pi/N)|}{|W(-\delta 2\pi/N - 2\pi/N)| + 2|W(-\delta 2\pi/N)| + |W(-\delta 2\pi/N + 2\pi/N)|} \end{aligned} \tag{44}$$

as

$$|V(\omega_0)| = \frac{2\pi}{\sin(\delta\pi)} \frac{|V_{k-1}| + 2|V_k| + |V_{k+1}|}{S_{k-1} + 2S_k + S_{k+1}}. \tag{45}$$

The phase of the frequency bin ω_0 for two-point and three-point interpolation is estimated the same way as for rectangular window (28) and Hanning window (39)

$$\arg\{V(\omega_0)\} = \arg\{V(\omega_k)\} \pm \arg\{e^{-j\delta(\pi/N)(N-1)}\}. \tag{46}$$

For higher order RVCI windows better accuracy in phase estimation may be obtained by replacing angle correction, that is second term in (46), by the angle of the following frequency bin

$$W(e^{j\omega}) \approx$$
$$\approx e^{-j\omega(N-1)/2}\left(\sum_{m=-M}^{M}(-1)^n A'[|m|](1+jm\pi/N)\frac{\sin(\omega N/2)}{\sin(\omega/2+m\pi/N)}\right), \quad (47)$$
$$\begin{cases}\text{for } m=0 \; A'_m = A_0^w, \\ \text{for } m\neq 0 \; A'_m = A_{|m|}^w/2.\end{cases}$$

For known frequency ω_0 amplitude and phase of the signal may also be computed from the definition of Fourier transform (4) or in the least squares (LS) sense.

3.2 Sinusoidal signal – arbitrary windows

IpDFT formulas (41), (43) are only valid for RVCI windows, which have wide main lobe that deteriorates frequency resolution and noise performance. In practice it is often desired to analyze the signal with the window having better properties. It is known from literature, e.g. (Harris, 1978), that optimal parametric non cosine Kaiser-Bessel and Dolph-Chebyshev windows often perform superior over other windows including RVCI windows. In the following part IpDFT algorithm for arbitrary, even non cosine, windows is described for two- and three-point interpolation. The algorithm is based on polynomial approximation of the ratio $R(\delta)$ (15), (17) for the window of interest.

First, the ratio $R(\delta)$ (15), (17) is computed numerically for the selected window based on window spectrum, and then the dependence $\delta=f_\delta(R)$ is approximated by polynomial. During analysis the ratio $R(\delta)$ (15), (17) is evaluated from DFT bins, and frequency correction is estimated from previously computed $\delta=f_\delta(R)$.

From the LS polynomial approximation we get

$$\delta \approx P_d(R), \quad P_d(R)=\sum_{l=0}^{L} a_l R^l , \quad (48)$$

where P_d denotes L degree polynomial, and R is computed from window's spectrum. Signal's frequency is given by

$$\omega_0 = [k \pm P_d(R)](2\pi/N) , \quad (49)$$

but this time R is computed from DFT bins of the analyzed signal.

Signal's amplitude is determined from the dependence $X_N=f_X(\delta)$, where for two-point interpolation

$$X_N = \frac{|V(\omega_0)|}{|V_k|} \approx \frac{|W(0)|}{|W(\delta 2\pi/N)|}, \quad (50)$$

and for three-point interpolation

$$X_N = \frac{|V(\omega_0)|}{|V_{k-1}|+2|V_k|+|V_{k+1}|} \approx$$
$$\approx \left(\frac{|W(-2\pi/N)|+2|W(0)|+|W(2\pi/N)|}{|W(-\delta 2\pi/N-2\pi/N)|+2|W(-\delta 2\pi/N)|+|W(-\delta 2\pi/N+2\pi/N)|} \right) / \quad (51)$$
$$/(2+2|W(2\pi/N)|/|W(0)|).$$

Ratios (50), (51) are next approximated by polynomial P_x

$$X_N \approx P_x(\delta). \quad (52)$$

Signal's amplitude for two-point interpolation is

$$|V(\omega_0)| = P_x(\delta)|V_k|, \quad (53)$$

and signal's amplitude for three-point interpolation is

$$|V(\omega_0)| = P_x(\delta)(|V_{k-1}|+2|V_k|+|V_{k+1}|). \quad (54)$$

Signal's phase is computed the same way for two-point and three-point interpolation based on the dependence $P_N=f_P(\delta)$

$$P_N = \arg\{W(\delta 2\pi/N)\}. \quad (55)$$

Signal's phase is

$$\arg\{V(\omega_0)\} = \arg\{V_k\} \pm P_p(\delta), \quad (56)$$

where P_p is polynomial approximating $P_N=f_P(\delta)$, i.e. $P_N \approx P_p(\delta)$.

Fig. 6 shows the dependence $\delta=f_\delta(R)$ for RVCI windows for two-point and three-point interpolation. Fig. 7 presents systematic errors of frequency estimation in dependence of the order of approximation polynomial for selected RVCI, Kaiser-Bessel and Dolph-Chebyshev windows for signal's frequencies ω_0=0.05 rad and ω_0=1 rad and signal's length N=512.

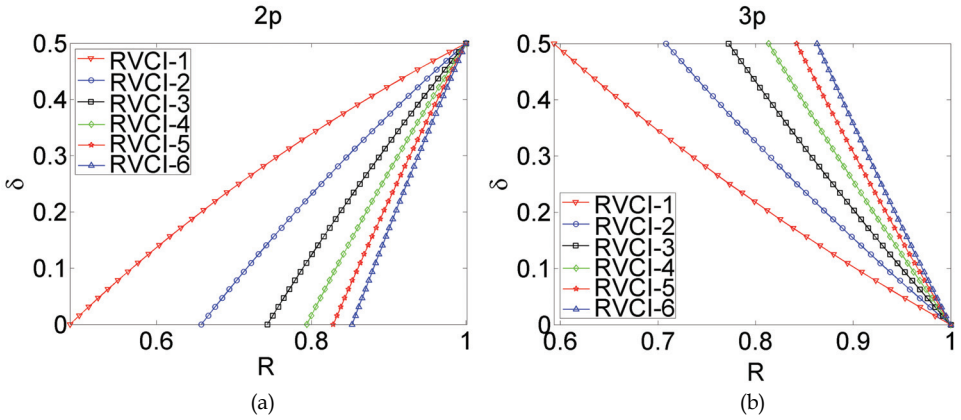


Fig. 6. Dependence $\delta=f_\delta(R)$ for RVCI windows for: a) two-point and b) three-point interpolation

Approximation polynomial was fitted to 64 points of window spectrum computed numerically via Fourier transform (4) in the frequency range from 0 to $2\pi/N$ rad, i.e. $W(e^{-j\omega})$ was computed by (4) for the set of frequencies from ω=0 to $\omega=2\pi/N$ rad with the increment $(2\pi/N)/63$ rad. Systematic errors were defined similar to (Schoukens et al., 1992), for each frequency ω_0, test signals were generated with the phase from the interval $<-\pi/2, \pi/2>$ changed with the step $\pi/20$ and the maximum absolute value of differences between estimated and true frequency was selected. It is seen from Fig. 7 that approximation polynomial of order 5 may give acceptable small systematic errors, nevertheless, in results shown in section 4 approximation polynomial of order 10 is used. For that order systematic errors, in Matlab 64 bit precision, for RVCI windows are practically the same for analytic IpDFT formulas (41), (43) and described approximation based IpDFT (49).

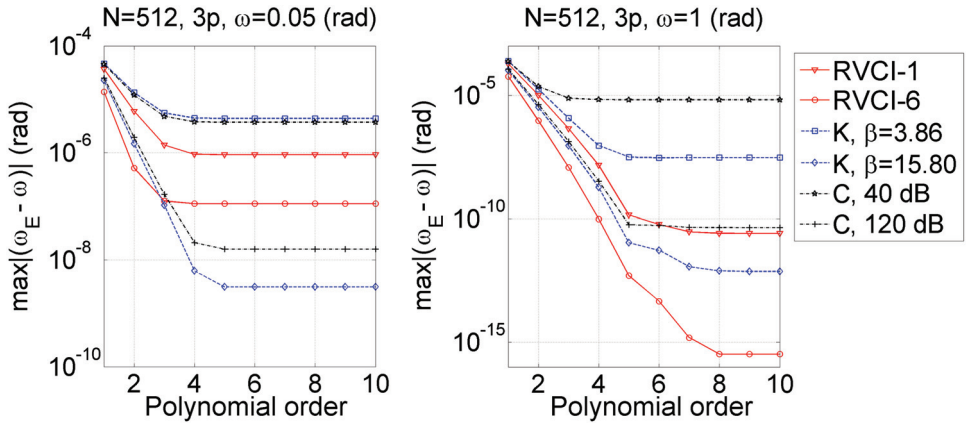


Fig. 7. Systematic errors of frequency estimation for 3p IpDFT for selected RVCI, Kaiser-Bessel and Dolph-Chebyshev windows for: a) ω_0=0.05 rad, and b) ω_0=1 rad

3.3 Damped sinusoidal signal – Bertocco-Yoshida algorithms

We start this section with derivation of DFT for damped sinusoidal signal (2). Let us rewrite signal (2) in the complex form

$$v_n = A\cos(\omega_0 n+\varphi)e^{-dn} = \frac{A}{2}\left(e^{-dn}e^{j(\omega_0 n+\varphi)} + e^{-dn}e^{-j(\omega_0 n+\varphi)}\right). \tag{57}$$

From the definition (5) DFT of the first term in the sum (57) is

$$\mathrm{DFT}\{e^{-dn}e^{j(\omega_0 n+\varphi)}\} = \sum_{n=0}^{N-1} e^{-dn}e^{j(\omega_0 n+\varphi)}e^{-j(2\pi/N)kn} = e^{j\varphi}\sum_{n=0}^{N-1} e^{(j\omega_0 - j\omega_k - d)n}, \tag{58}$$

where $\omega_k=(2\pi/N)k$. Using the formula for the sum of the geometric series (58) becomes

$$\mathrm{DFT}\{e^{-dn}e^{j(\omega_0 n+\varphi)}\} = e^{j\phi}\frac{1-e^{(j\omega_0 - j\omega_k - d)N}}{1-e^{j\omega_0 - j\omega_k - d}}. \tag{59}$$

Going the same way for the second term in (57) we finally obtain the DFT of the damped sinusoidal signal (2)

$$V_k = \frac{A}{2}\left(e^{j\varphi}\frac{1-e^{(j\omega_0 - j\omega_k - d)N}}{1-e^{j\omega_0 - j\omega_k - d}} + e^{-j\phi}\frac{1-e^{(-j\omega_0 - j\omega_k - d)N}}{1-e^{-j\omega_0 - j\omega_k - d}} \right), \tag{60}$$

which may be rewritten considering that $e^{-j\omega_k N} = 1$ and $\lambda = e^{-d+j\omega_0}$ in the form

$$V_k = \frac{A}{2}\left(e^{j\varphi}\frac{1-\lambda^N}{1-\lambda e^{-j\omega_k}} + e^{-j\varphi}\frac{1-\lambda^{*N}}{1-\lambda^* e^{-j\omega_k}} \right). \tag{61}$$

In the following derivation of Bertocco-Yoshida IpDFT algorithms it is assumed that

$$V_k \approx \frac{A}{2}\left(e^{j\varphi}\frac{1-\lambda^N}{1-\lambda e^{-j\omega_k}} \right), \quad 0 \le \omega < \pi \,. \tag{62}$$

3.3.1 Bertocco (BY-0) algorithm

Let us define the following ratio of complex DFT bins

$$R = \frac{V_{k+1}}{V_k} \approx \frac{\frac{A}{2}\left(e^{j\varphi}\frac{1-\lambda^N}{1-\lambda e^{-j\omega_{k+1}}} \right)}{\frac{A}{2}\left(e^{j\varphi}\frac{1-\lambda^N}{1-\lambda e^{-j\omega_k}} \right)} = \frac{1-\lambda e^{-j\omega_k}}{1-\lambda e^{-j\omega_{k+1}}}, \tag{63}$$

where V_k is the DFT bin with the highest magnitude. From (63) we get

$$\lambda = e^{j\omega_k}\frac{1-R}{1-Re^{-j2\pi/N}} \,. \tag{64}$$

Considering $\lambda = e^{-d+j\omega_0}$ damping and frequency are given by

$$d = -\mathrm{Re}\{\ln(\lambda)\}\,,\ \omega_0 = \mathrm{Im}\{\ln(\lambda)\}\,. \tag{65}$$

3.3.2 BY-1 algorithm

Let us define the ratio of the first order differences of the complex DFT bins in the form

$$R = \frac{V_{k-1} - V_k}{V_k - V_{k+1}} \,. \tag{66}$$

where V_k is the DFT bin with the highest magnitude. Substituting (62) into (66) we get

$$R = \frac{V_{k-1} - V_k}{V_k - V_{k+1}} = \frac{1-\lambda e^{-j\omega_{k+1}}}{1-\lambda e^{-j\omega_{k-1}}} r, \quad r = \frac{-e^{-j\omega_k} + e^{-j\omega_{k-1}}}{-e^{-j\omega_{k+1}} + e^{-j\omega_k}}, \tag{67}$$

$$\lambda = e^{j\omega_k}\frac{r-R}{re^{-j2\pi/N} - Re^{j2\pi/N}} \,. \tag{68}$$

Damping and frequency are given by (65).

3.3.3 Yoshida (BY-2) algorithm

Let us define the ratio of the second order differences of the complex DFT bins in the form

$$R = \frac{V_{k-2} - 2V_{k-1} + V_k}{V_{k-1} - 2V_k + V_{k+1}} = \frac{1 - \lambda e^{-j\omega_{k+1}}}{1 - \lambda e^{-j\omega_{k-2}}} r \,, \tag{69}$$

where

$$r = r_1 / r_2$$
$$r_1 = (1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_k}) - 2(1 - \lambda e^{-j\omega_{k-2}})(1 - \lambda e^{-j\omega_k}) + (1 - \lambda e^{-j\omega_{k-2}})(1 - \lambda e^{-j\omega_{k-1}})$$
$$r_2 = (1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+1}}) - 2(1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_{k+1}}) + (1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_k})$$

and λ is evaluated from (68). From (69) we get

$$\lambda = e^{j\omega_k} \frac{1 - R}{e^{-j2\pi/N} - R e^{j2(2\pi/N)}} \,. \tag{70}$$

Damping and frequency are given by (65).

Second order differences may also be defined as

$$R = \frac{V_{k-1} - 2V_k + V_{k+1}}{V_k - 2V_{k+1} + V_{k+2}} = \frac{1 - \lambda e^{-j\omega_{k+2}}}{1 - \lambda e^{-j\omega_{k-1}}} r \,, \tag{71}$$

where

$$r = r_1 / r_2$$
$$r_1 = (1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+1}}) - 2(1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_{k+1}}) + (1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_k})$$
$$r_2 = (1 - \lambda e^{-j\omega_{k+1}})(1 - \lambda e^{-j\omega_{k+2}}) - 2(1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+2}}) + (1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+1}})$$

and λ is evaluated from (68). From (71) we get

$$\lambda = e^{j\omega_k} \frac{1 - R}{e^{-j2(2\pi/N)} - R e^{j2\pi/N}} \,. \tag{72}$$

Damping and frequency are given by (65).

Fig. 8 illustrates cases for definitions of the ratio (69) and (71). Four successive DFT bins are always taken for interpolation and the DFT bin V_k has the highest magnitude. For (69) DFT bins with the highest magnitudes are k-2, k-1, k, k+1 (Fig. 8a) and for (71) k-1, k, k+1, k+2 (Fig. 8b).

In the original derivation of Yoshida algorithm the ratio (69) is used and damping and frequency are given by

$$d = \frac{2\pi}{N} \mathrm{Im}\{-3 / (R-1)\} \,, \;\; \omega = \frac{2\pi}{N} \mathrm{Re}\{k - 3 / (R-1)\} \,. \tag{73}$$

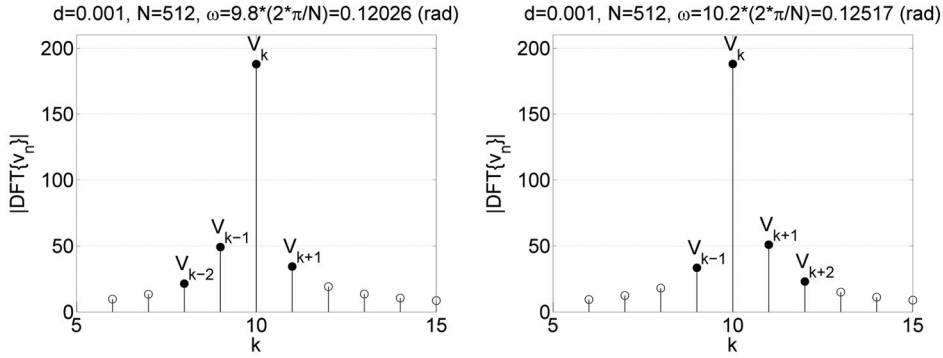


Fig. 8. DFT of the damped sinusoidal signal. Solid circles denote DFT bins taken for Yoshida (BY-2) algorithm: a) ratio defined by (69), b) ratio defined by (71)

3.3.4 BY-3 algorithm

Let us define the ratio of the third order differences of the complex DFT bins in the form

$$R = \frac{V_{k-2} - 3V_{k-1} + 3V_k - V_{k+1}}{V_{k-1} - 3V_k + 3V_{k+1} - V_{k+2}} = \frac{1 - \lambda e^{-j\omega_{k+2}}}{1 - \lambda e^{-j\omega_{k-2}}} r \,, \tag{74}$$

where

$$\begin{aligned} r &= r_1 / r_2 \\ r_1 &= (1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+1}}) - 3(1 - \lambda e^{-j\omega_{k-2}})(1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+1}}) \\ &\quad + 3(1 - \lambda e^{-j\omega_{k-2}})(1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_{k+1}}) - (1 - \lambda e^{-j\omega_{k-2}})(1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_k}) \\ r_2 &= (1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+1}})(1 - \lambda e^{-j\omega_{k+2}}) - 3(1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_{k+1}})(1 - \lambda e^{-j\omega_{k+2}}) \\ &\quad + 3(1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+2}}) - (1 - \lambda e^{-j\omega_{k-1}})(1 - \lambda e^{-j\omega_k})(1 - \lambda e^{-j\omega_{k+1}}) \end{aligned}$$

and λ is evaluated from (68). From (74) we get

$$\lambda = e^{j\omega_k} \frac{r - R}{re^{-j2(2\pi/N)} - Re^{j2(2\pi/N)}} \,. \tag{75}$$

Damping and frequency are given by (65).

3.4 Damped sinusoidal signal – RVCI windows

In the derivation of IpDFT algorithms for RVCI windows we treat damped signal with RVCI window as sinusoidal signal with damped window i.e.

$$v_n = w_n x_n = w_n A\cos(\omega_0 n + \varphi)e^{-dn} = \overline{w}_n A\cos(\omega_0 n + \varphi) \,, \tag{76}$$

where $\overline{w}_n = w_n e^{-dn}$ is damped time window.

3.4.1 Rectangular window (RVCI *M*=0)

Based on the spectrum of rectangular window (12) the spectrum of the damped rectangular window is

$$\bar{W}^R(e^{j\omega},d)=\sum_{n=0}^{N-1}e^{-dn}e^{-j\omega n}=\sum_{n=0}^{N-1}e^{-j(\omega-jd)n}=e^{-j\bar{\omega}(N-1)/2}\frac{\sin(\bar{\omega}N/2)}{\sin(\bar{\omega}/2)}=\bar{W}^R(e^{j\bar{\omega}}), \tag{77}$$

where $\bar{\omega}=\omega-jd$. Let us define following ratios of the squares of frequency bins of the damped rectangular window spectrum

$$R_1=\frac{|\bar{W}^R(\bar{\omega}_{k+1})|^2}{|\bar{W}^R(\bar{\omega}_k)|^2},\quad R_2=\frac{|\bar{W}^R(\bar{\omega}_{k-1})|^2}{|\bar{W}^R(\bar{\omega}_k)|^2}, \tag{78}$$

where $|\bar{W}^R(\bar{\omega}_k)|$ is a frequency bin with the highest modulus, see Fig. 5. Using (77) the ratios (78) are

$$R_1=\frac{|\bar{W}^R(\bar{\omega}_{k+1})|^2}{|\bar{W}^R(\bar{\omega}_k)|^2}\approx\frac{|V_{k+1}|^2}{|V_k|^2}\approx\frac{|\delta+jD|^2}{|\delta-1+jD|^2}=\frac{\delta^2+D^2}{(\delta-1)^2+D^2}, \tag{79}$$

$$R_2=\frac{|\bar{W}^R(\bar{\omega}_{k-1})|^2}{|\bar{W}^R(\bar{\omega}_k)|^2}\approx\frac{|V_{k-1}|^2}{|V_k|^2}\approx\frac{|\delta+jD|^2}{|\delta+1+jD|^2}=\frac{\delta^2+D^2}{(\delta+1)^2+D^2}, \tag{80}$$

where $D=dN/(2\pi)$. In derivation of (79-80) sine functions were approximated by theirs arguments and it was assumed that $(N-1)/N\approx 1$.

By comparing D^2 in (79) and (80) we get desired frequency correction defined by (13)

$$\delta=-\frac{1}{2}\frac{R_1-R_2}{2R_1R_2-R_1-R_2}. \tag{81}$$

The damping computed from (79) is

$$d=\frac{2\pi}{N}\sqrt{\frac{\delta^2-R_1(\delta-1)^2}{R_1-1}},\quad \delta\neq 0.5, \tag{82}$$

and damping computed from (80) is

$$d=\frac{2\pi}{N}\sqrt{\frac{\delta^2-R_2(\delta+1)^2}{R_2-1}},\quad \delta\neq -0.5. \tag{83}$$

For δ=0.5 (82) must not be used. In implementation if δ=0.5 then zero sample should be appended at the end of the signal to change δ. Equation (83) may always be used, as from definition (13) frequency correction is never equal -0.5.

Modulus of the frequency bin ω_0 may be computed from the proportion

$$\frac{|V(\omega_0)|}{|V_k|}=\frac{|\bar{W}^R(0)|}{|\bar{W}^R(\bar{\omega}_k)|}=\frac{|\bar{W}^R(-jd)|}{|\bar{W}^R(-\delta 2\pi/N-jd)|}. \tag{84}$$

From (84) we get

$$|V(\omega_0)| = |V_k| \frac{\sqrt{\delta^2 + D^2}}{D} \frac{|\sin(jdN/2)|}{|\sin(\delta\pi + jdN/2)|}, \quad d \neq 0, \tag{85}$$

and signal's amplitude is

$$A = |V(\omega_0)| / e^{-d(N-1)/2}. \tag{86}$$

Signal's phase may be computed as in case of sinusoidal signals (27)

$$\arg\{V(\omega_0)\} = \arg\{V(\omega_k)\} \pm \arg\{\bar{W}^R(\delta 2\pi / N - jd)\}. \tag{87}$$

Sign '+' or '-' in (87) is taken the same way as in (13).

3.4.2 Hanning (Hann, RVCI *M*=1) window

The spectrum of damped Hanning window $\bar{w}_n^H = w_n^H e^{-dn}$ is given by

$$\bar{W}^H(e^{j\bar{\omega}}) = -0.25\bar{W}^R(e^{j(\bar{\omega}-\omega_1)}) + 0.5\bar{W}^R(e^{j\bar{\omega}}) - 0.25\bar{W}^R(e^{j(\bar{\omega}+\omega_1)}), \quad \omega_1 = 2\pi/N, \tag{88}$$

where $\bar{W}^R(e^{j\bar{\omega}})$ is the spectrum of damped rectangular window (77) and $\bar{\omega} = \omega - jd$. Inserting (77) into (88) and assuming $e^{j(\pi/N)(N-1)} \approx -1 + j\pi/N$ and π/N<<1 the spectrum of damped Hanning window is further approximated by

$$\bar{W}^H(e^{j\bar{\omega}}) \approx e^{-j\bar{\omega}(N-1)/2} \sin(\bar{\omega}N/2)\left(-\frac{0.25}{\sin(\bar{\omega}/2 - \pi/N)} + \frac{0.5}{\sin(\bar{\omega}/2)} - \frac{0.25}{\sin(\bar{\omega}/2 + \pi/N)}\right). \tag{89}$$

The ratios of the squares of frequency bins of the damped Hanning window spectrum are

$$R_1 = \frac{|\bar{W}(\bar{\omega}_{k+1})|^2}{|\bar{W}(\bar{\omega}_k)|^2} \approx \frac{|V_{k+1}|^2}{|V_k|^2} \approx \frac{(\delta+1)^2 + D^2}{(\delta-2)^2 + D^2}, \tag{90}$$

$$R_2 = \frac{|\bar{W}(\bar{\omega}_{k-1})|^2}{|\bar{W}(\bar{\omega}_k)|^2} \approx \frac{|V_{k-1}|^2}{|V_k|^2} \approx \frac{(\delta-1)^2 + D^2}{(\delta+2)^2 + D^2}, \tag{91}$$

where $D=dN/(2\pi)$. From (90) and (91) we get desired frequency correction defined by (13)

$$\delta = -\frac{3}{2}\frac{R_1 - R_2}{4R_1R_2 - R_1 - R_2 - 2}. \tag{92}$$

The damping computed from (90) and (91) are

$$d = \frac{2\pi}{N}\sqrt{\frac{(\delta+1)^2 - R_1(\delta-2)^2}{R_1 - 1}}, \quad \delta \neq 0.5, \; d = \frac{2\pi}{N}\sqrt{\frac{(\delta-1)^2 - R_2(\delta+2)^2}{R_2 - 1}}, \quad \delta \neq -0.5 \tag{93}$$

Signal's amplitude and phase are next computed as

$$|V(\omega_0)|=|V_k|\,|\bar{W}^H(0)|/|\bar{W}^H(\bar{\omega}_k)|,\ A=|V(\omega_0)|/e^{-d(N-1)/2}, \tag{94}$$

$$\arg\{V(\omega_0)\}=\arg\{V(\omega_k)\}\pm\arg\{\bar{W}^H(\delta 2\pi/N-jd)\}. \tag{95}$$

3.4.3 Higher order RVCI windows

In general, the spectrum of the damped $\bar{w}_n=w_n e^{-dn}$ RVCI window order M is a sum of rescaled and moved in frequency spectra of damped rectangular window

$$\bar{W}_M(e^{j\bar{\omega}})=\sum_{m=0}^{M}(-1)^m\frac{A_m^w}{2}\bar{W}^R(e^{j(\bar{\omega}-\omega_m)})+(-1)^m\frac{A_m^w}{2}\bar{W}^R(e^{j(\bar{\omega}+\omega_m)}). \tag{96}$$

For damped RVCI window the ratios of the squares of frequency bins are

$$R_1=\frac{|\bar{W}_M(\bar{\omega}_{k+1})|^2}{|\bar{W}_M(\bar{\omega}_k)|^2}\approx\frac{|V_{k+1}|^2}{|V_k|^2}\approx\frac{(\delta+M)^2+D^2}{(\delta-M-1)^2+D^2}, \tag{97}$$

$$R_2=\frac{|\bar{W}_M(\bar{\omega}_{k-1})|^2}{|\bar{W}_M(\bar{\omega}_k)|^2}\approx\frac{|V_{k-1}|^2}{|V_k|^2}\approx\frac{(\delta-M)^2+D^2}{(\delta+M+1)^2+D^2}, \tag{98}$$

where $D=dN/(2\pi)$ and M is the order of RVCI window. For M=0 (rectangular window) (97-98) becomes (79-80), and for M=1 (Hanning window) (97-98) is (90-91).

By comparing D^2 in (97) and (98) we get general formula for frequency correction defined by (13) for damped RVCI order M window

$$\delta=-\frac{2M+1}{2}\frac{R_1-R_2}{2(M+1)R_1R_2-R_1-R_2-2M}. \tag{99}$$

The damping computed from (97) and (98) are

$$d=\frac{2\pi}{N}\sqrt{\frac{(\delta+M)^2-R_1(\delta-M-1)^2}{R_1-1}},\ \delta\neq 0.5,\ d=\frac{2\pi}{N}\sqrt{\frac{(\delta-M)^2-R_2(\delta+M+1)^2}{R_2-1}},\ \delta\neq -0.5. \tag{100}$$

Signal's amplitude and phase are next computed as

$$|V(\omega_0)|=|V_k|\,|\bar{W}(0)|/|\bar{W}(\bar{\omega}_k)|,\ A=|V(\omega_0)|/e^{-d(N-1)/2}, \tag{101}$$

$$\arg\{V(\omega_0)\}=\arg\{V(\omega_k)\}\pm\arg\{\bar{W}(\delta 2\pi/N-jd)\}. \tag{102}$$

4. Some properties of IpDFT algorithms

In this section we present results of simulations that describe systematic errors and noise immunity of IpDFT methods. Because of space constrains, only the results of frequency or frequency and damping estimation are presented. Including results for amplitude and phase estimation would multiply the number of figures by three. Furthermore, in practice

estimation of frequency and damping is of primary importance, and once having frequency and damping the amplitude and phase may be estimated by LS or FT. It is also true that amplitude and phase estimation errors, not shown in this section, behave similarly to frequency estimation errors.

First, systematic errors of IpDFT algorithms for sinusoidal and damped sinusoidal signals are presented, and then robustness against additive, zero-mean, Gaussian noise is shown. In all simulations number of samples N=512 was chosen.

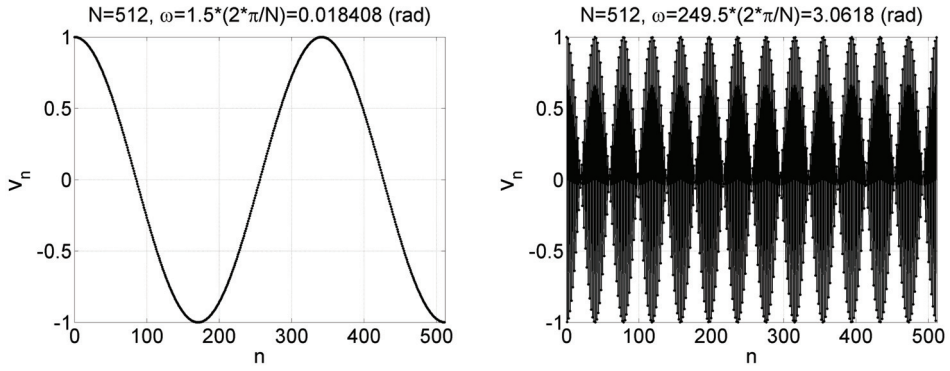
Simulations were conducted in Matlab 64-bit floating point precision. Accuracy of this precision determined by the function *eps* (Matlab) is on the level 10^{-15}–10^{-16}, and estimation errors cannot be lower than this accuracy.

4.1 Systematic errors

In this section systematic errors of frequency estimation for sinusoidal signals and frequency and damping estimation for damped sinusoidal signals are presented. For each frequency ω_0 or damping d, test signals were generated with the phase from the interval $<-\pi/2, \pi/2>$ changed with the step $\pi/20$ and the maximum absolute difference between estimated and true value was selected.

For obtaining general conclusions the frequency of the test signals was swept in the whole range from 0 to π rad. For easier interpretation the frequency of the test signal is also given in DFT index k.

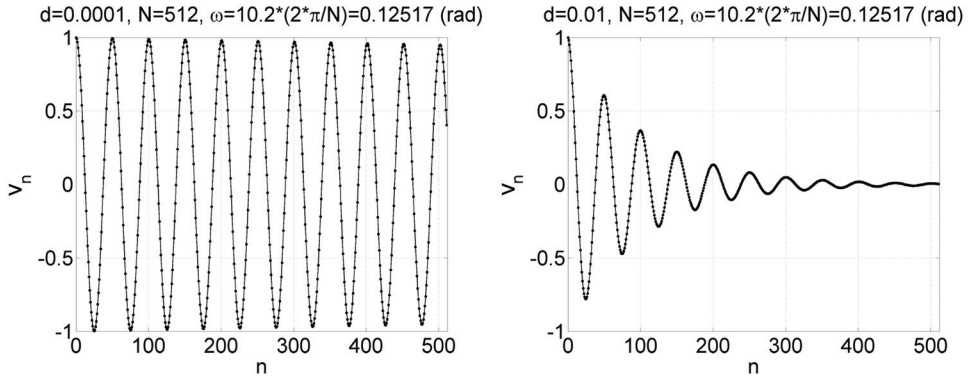
Fig. 9 shows two sinusoidal test signals with N=512 samples. The signal in Fig. 9a with frequency $\omega_0=1.5\cdot(2\pi/N)$ contains 1.5 periods, whereas signal in Fig. 9b with frequency $\omega_0=249.5\cdot(2\pi/N)$ rad contains 249.5 periods. Frequencies of those signals scaled in DFT index k are 1.5 and 249.5 and it means, that in the frequency spectrum those signals lie in the half way between DFT bins k=1 and k=2 and bins k=249 and k=250, respectively, and in both cases frequency correction δ (13) equals 0.5. The first signal is sampled approx 341 times per period and the second only approx 2.05 times per period.



(a) (b)

Fig. 9. Sinusoidal test signals N=512, a) signal with frequency $\omega_0=1.5\cdot(2\pi/N)$ rad containing 1.5 periods, b) signal with frequency $\omega_0=249.5\cdot(2\pi/N)$ rad containing 249.5 periods

Fig. 10 depicts damped sinusoidal test signals with frequency ω_0=10.2($2\pi/N$) rad, i.e. δ=0.2. In simulations the range of damping from d=0.0001 (Fig.10a) to d=0.01 (Fig.10b) is considered.



(a) (b)

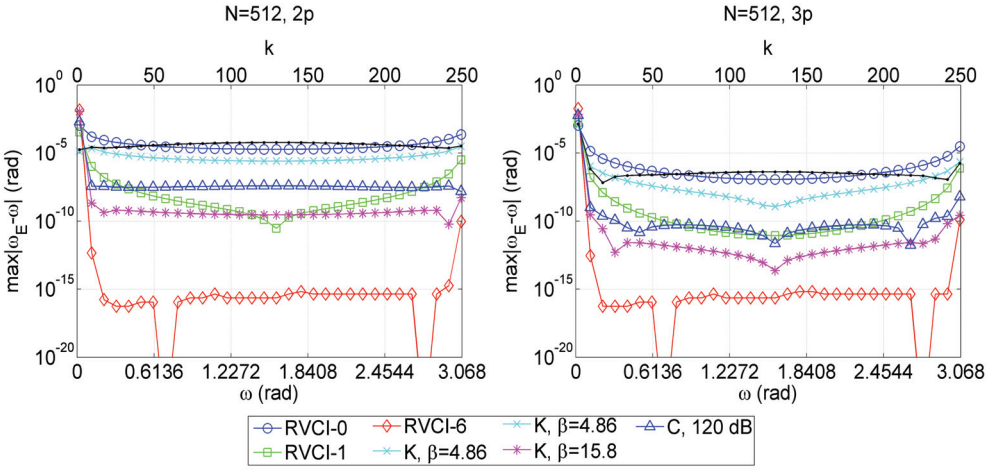
Fig. 10. Damped sinusoidal test signals N=512, ω_0=10.2 ·($2\pi/N$) rad, a) d=0.0001, b) d=0.01

Figs. 11-12 present systematic errors of sinusoidal signal frequency estimation for selected RVCI, Kaiser-Bessel and Dolph-Chebyshev windows for two-point and three-point interpolation. The frequency of the test signals in Fig. 11 was changed from ω_0=1.5 ·($2\pi/N$) to ω_0=249.5 ·($2\pi/N$) with the step 8 ·($2\pi/N$), and the frequency of the test signals in Fig. 12 was changed from ω_0=2 ·($2\pi/N$) to ω_0=6 ·($2\pi/N$) with the step ($2\pi/N$)/5. In the first case test signal was newer coherently sampled (i.e. signal never contained integer number of periods), whereas in the second case coherent sampling occurred for test signals containing exactly 2, 3, 4, 5, 6 periods.

It is seen from Figs. 11-12 that 3p interpolation gives smaller systematic errors than 2p interpolation. Increasing the order of RVCI window results in significant reduction of systematic errors. High order RVCI M=6 window may give negligible small estimation errors as visible in Fig. 11 which is the effect of the fastest decay of the sidelobes. Still, due to wide main lobe of RVCI M=6 window, the signal has to contain sufficient number of periods. Systematic errors for Kaiser-Bessel and Dolph-Chebyshev windows are significantly higher then for RVCI M=6 window because sidelobes of those windows does not decay so fast. However, it is seen from Fig. 12 that for analysis of the short signal containing 2-4 periods it is advantageous to use, narrow main lobe, Dolph-Chebyshev 120 dB and Kaiser-Bessel β=15.8 windows over RVCI windows. Local minima for integer values of k in Fig. 12 occur for coherent sampling and cosine windows. For the case of coherent sampling frequency correction is δ=0, DFT analysis is correct and there is no need to use IpDFT algorithms.

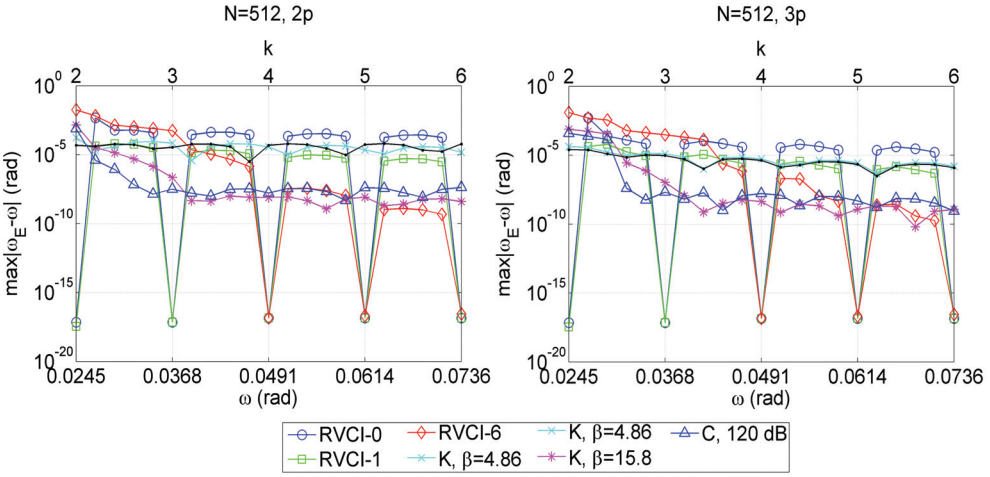
Figs. 13-14 present systematic errors of damped sinusoidal signal frequency and damping estimation for BY algorithms and IpDFT with RVCI windows. The frequency of the test signals in Fig. 13 was changed from ω_0=1.5 ·($2\pi/N$) to ω_0=249.5 ·($2\pi/N$) with the step 8 ·($2\pi/N$), and damping was set to d=0.01 (compare Fig. 10b). In Fig. 14 the damping was

swept from d=10^{-4} to 10^{-2} with equidistant steps in logarithmic scale, and the frequency was set to ω_0=10.2·($2\pi/N$) (i.e. δ=0.2). It is seen from Figs. 13-14 that by choosing high order RVCI windows significant reduction of systematic errors may be obtained, however, the price for this gain is the need for longer signal (in the sense of number of cycles) as explain previously for sinusoidal signals, and higher noise sensitivity as shown in the next section.



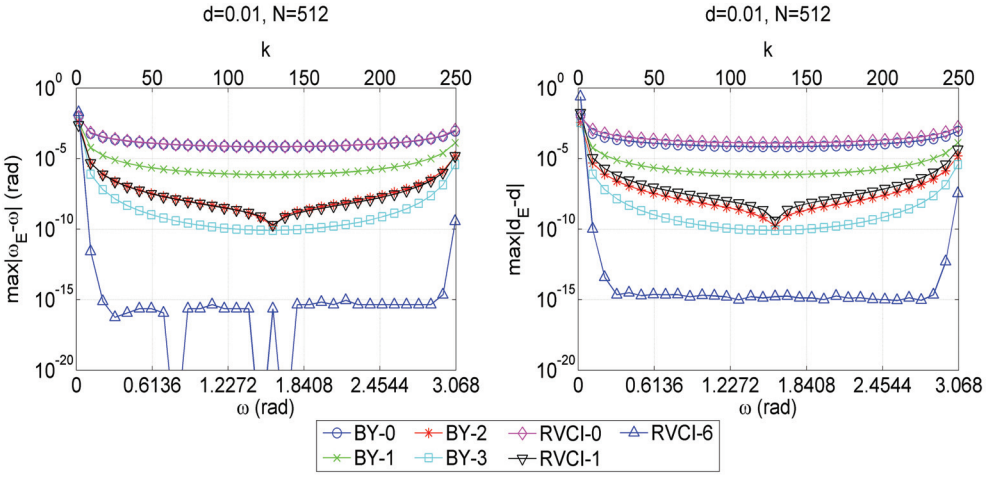
(a) (b)

Fig. 11. Systematic errors of sinusoidal signal frequency estimation for selected RVCI, Kaiser-Bessel and Dolph-Chebyshev windows: a) 2p interpolation, b) 3p interpolation



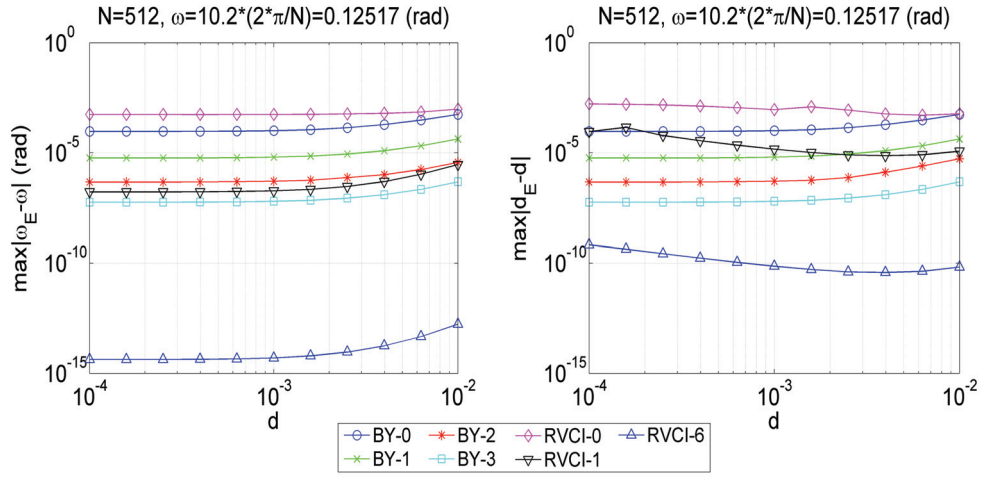
(a) (b)

Fig. 12. Systematic errors of sinusoidal signal frequency estimation for selected RVCI, Kaiser-Bessel and Dolph-Chebyshev windows: a) 2p interpolation, b) 3p interpolation



(a) (b)

Fig. 13. Systematic errors of damped sinusoidal signal frequency and damping estimation for BY algorithms and RVCI windows



(a) (b)

Fig. 14. Systematic errors of damped sinusoidal signal frequency and damping estimation for BY algorithms and RVCI windows

4.2 Noise

Noise performance is typically illustrated by comparison with Cramér-Rao Lower Bound (CRLB). Unbiased estimator that reaches CRLB is optimal Minimum Variance Unbiased (MVU) estimator. CRLB for sinusoidal signal (1) disturbed by zero-mean Gaussian noise with variance σ^2 is given by (Kay, 1993)

$$\mathrm{var}(\omega_{0E}) \geq \frac{12}{\eta N(N^2-1)} \tag{103}$$

and for damped sinusoidal signal (2) by (Yao & Pandit, 1995)

$$\mathrm{var}(\omega_{0E}) = \mathrm{var}(d_E) \approx \frac{(1-z^2)^3(1-z^{2N})}{\eta[-N^2 z^{2N}(1-z^2)^2 + z^2(1-z^{2N})^2]}, \; z = |e^{-d+j\omega}|, \tag{104}$$

where E in subscripts stands for estimated value, and η is signal to noise ratio defined as

$$\eta = A^2/(2\sigma^2), \; S/N = 10\log_{10}(\eta) \quad (\mathrm{dB}). \tag{105}$$

It is seen from (103-104) that the variance of frequency estimator is inverse proportional to the third power of signal length N. This strong dependence suggests high number of samples for frequency estimation, i.e. high sampling frequency or/and long observation time.

Fig. 15 shows exemplary realization of the sinusoidal signal disturbed by the zero-mean Gaussian noise with S/N=10 dB (105) and its DFT spectrum. Mean value and standard deviation of estimation errors, shown in Figs. 16-18, were computed from 1000 realizations of the test signal. Signal's phase was generated as a random variable with uniform distribution on the interval <-π/2, π/2>.

Fig. 16 presents results of frequency estimation of sinusoidal signal analyzed with different windows for three-point interpolation. It is seen from Fig. 16 that for S/N from approx 15 dB to 40 dB rectangular window (RVCI M=0) has the best noise immunity. For the lower disturbance (higher S/N) systematic errors become more significant than noise for rectangular window and better results are obtained with RVCI M=1 (Hanning) window, Kaiser-Bessel β=4.86 window, and Dolph-Chebyshev 50 dB window. RVCI M=6 has the highest noise sensitivity due to the widest main lobe.

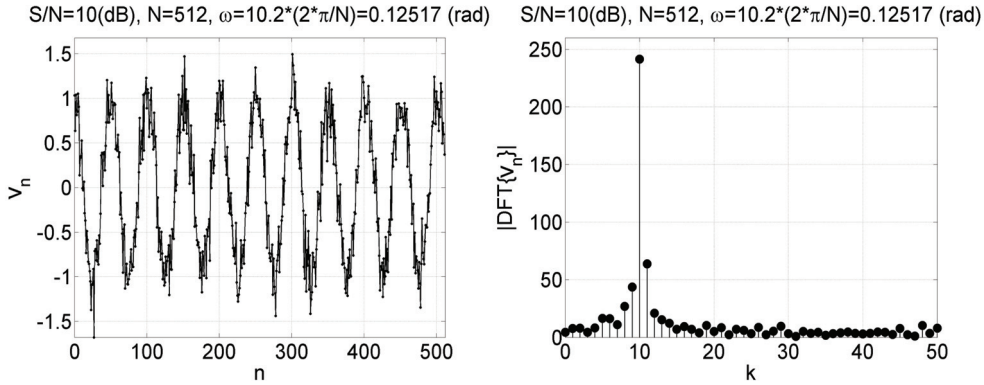
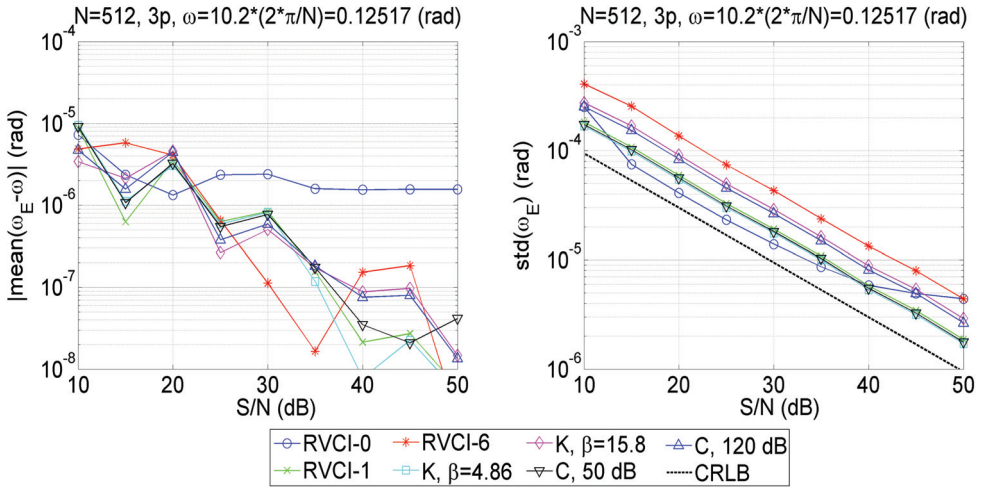


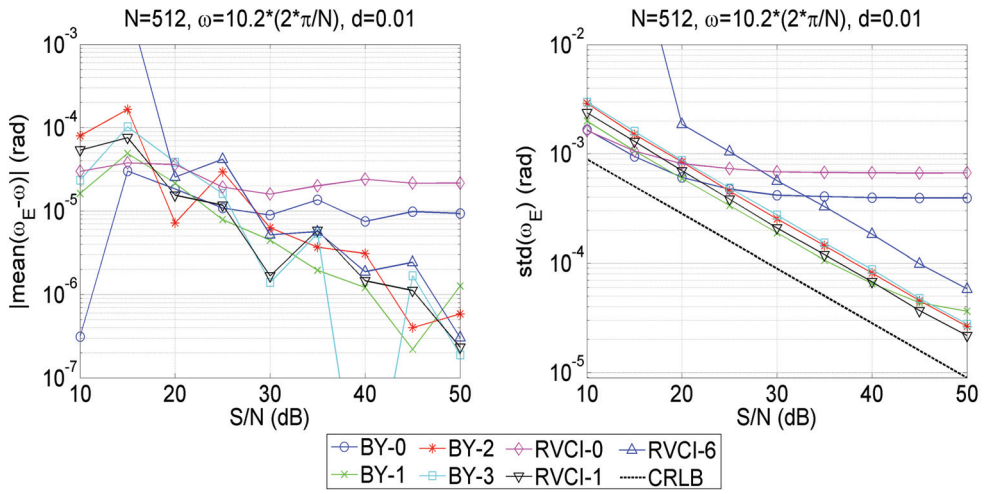
Fig. 15. a) Exemplary realization of sinusoidal test signal disturbed by 10 dB noise, N=512, ω_0=10.2·(2π/N) rad; b) DFT spectrum



(a) (b)

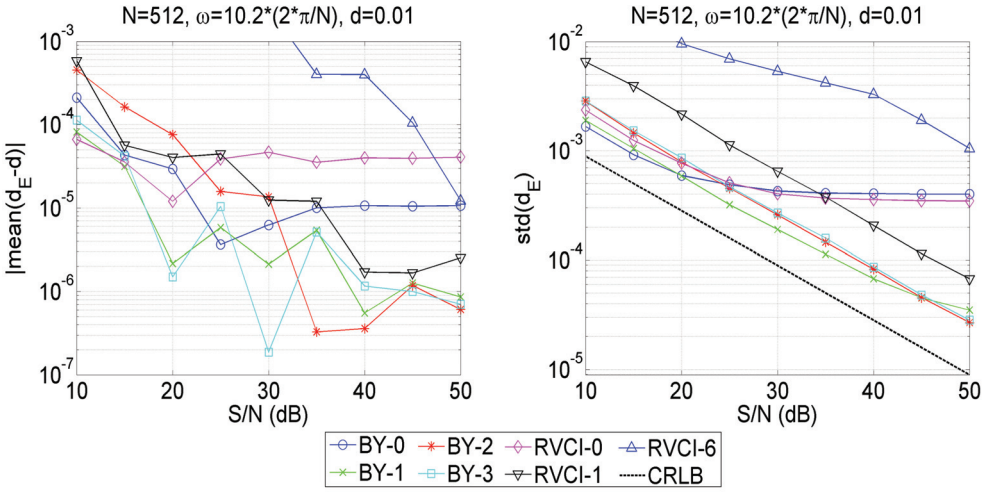
Fig. 16. a) Mean value and b) standard deviation of frequency estimation error for sinusoidal signal for different time windows

Figs. 17-18 present results of frequency and damping estimation for damped sinusoidal signal. It is seen that for high S/N systematic error is dominant for BY-0, and RVCI M=0 window. BY-1 algorithm gives the best results in the wide range of S/N. It is seen from Fig. 18b that BY methods are better estimators of damping than IpDFT with RVCI windows.



(a) (b)

Fig. 17. a) Mean value and b) standard deviation of frequency estimation error for damped sinusoidal signal for different IpDFT algorithms



(a) (b)

Fig. 18. a) Mean value and b) standard deviation of damping estimation error for damped sinusoidal signal for different IpDFT algorithms

5. Conclusion

This chapter describes DFT interpolation algorithms for parameters estimation of sinusoidal and damped sinusoidal signals. IpDFT algorithms have two main advantages:

1. Low computational complexity attributed to fast algorithms of DFT computation.
2. No need for the signal model (as opposed to parametric methods).

IpDFT methods may be used as fully functional estimators, especially when noise disturbance is not very strong. If the signal model is known IpDFT may be used for providing starting point for LS optimization that is optimal for Gaussian zero-mean noise disturbance.

For the signals with disturbances not possible to include in the signal model, as e.g. unknown drift, IpDFT with adequate time window may offer better performance than optimization.

6. References

Agrež, D. (2002) Weighted Multipoint Interpolated DFT to Improve Amplitude Estimation of Multifrequency Signal, *IEEE Trans. Instrum. Meas.*, vol. 51, pp. 287-292

Agrež, D. (2009) A frequency domain procedure for estimation of the exponentially damped sinusoids, *International Instrumentation and Measurement Technology Conference*, May 2009.

Andria, G., Savino, M. & Trotta, A. (1989) Windows and interpolation algorithms to improve electrical measurement accuracy, *IEEE Trans. Instrum. Meas.*, vol. 38, pp. 856–863

Bertocco, M., Offeli, C. & Petri, D. (1994) Analysis of damped sinusoidal signals via a frequency-domain interpolation algorithm, *IEEE Trans. Instrum. Meas.*, vol. 43, no. 2, pp.245-250

Cooley, J. W., Tukey, J. W. (1965) An Algorithm for the Machine Computation of Complex Fourier Series, *Mathematics of Computation*, vol. 19, pp. 297-301

Duda, K. (2010) Accurate, Guaranteed-Stable, Sliding DFT, *IEEE Signal Processing Mag.*, November, pp 124-127

Duda, K. (2011a) DFT Interpolation Algorithm for Kaiser-Bessel and Dolph-Chebyshev Windows, *IEEE Trans. Instrum. Meas.*, vol. 60, no. 3, pp. 784–790

Duda, K., Magalas, L. B., Majewski, M. & Zieliński, T. P. (2011b) DFT based Estimation of Damped Oscillation's Parameters in Low-frequency Mechanical Spectroscopy, *IEEE Trans. Instrum. Meas.*, vol. 60, no. 11, pp. 3608-3618

Grandke, T. (1983) Interpolation Algorithms for Discrete Fourier Transforms of Weighted Signals, *IEEE Trans. Instrum. Meas.*, vol. Im-32, No. 2, pp.350-355

Harris, F. J. (1978) On the use of windows for harmonic analysis with the discrete Fourier transform, *Proc. IEEE*, vol. 66, pp. 51–83

Jacobsen, E. & Lyons, R. (2003) The sliding DFT, *IEEE Signal Processing Mag.*, vol. 20, no. 2, pp. 74–80

Jain V. K., Collins, W. L. & Davis, D. C. (1979) High-Accuracy Analog Measurements via Interpolated FFT," *IEEE Trans. Instrum. Meas.*, vol. Im-28, No. 2, pp.113-122

Kay, S. M. (1993) *Fundamentals of Statistical Signal Processing: Estimation Theory*, Englewood Cliffs, NJ: Prentice-Hall

Lyons, R. G. (2004) *Understanding Digital Signal Processing*, Second Edition, Prentice-Hall

Matlab The Language of Technical Computing. Function Reference Volume 1–3, available from www.mathworks.com.

Moon, T. K. & Stirling W. C. (1999) *Mathematical Methods and Algorithms for Signal Processing*, Prentice Hall

Nuttall, A. H. (1981) Some Windows with Very Good Sidelobe Behavior, *IEEE Trans. On Acoustics, Speech, And Signal Processing*, vol. ASSP-29, no. 1, pp.84-91

Offelli, C. & Petri, D. (1990) Interpolation Techniques for Real-Time Multifrequency Waveform Analysis, *IEEE Trans. Instrum. Meas.*, Vol. 39. No. 1, pp.106-111

Oppenheim, A.V., Schafer, R.W. & Buck, J.R. (1999) *Discrete-Time Signal Processing*, 2nd Edition, Prentice-Hall

Radil, T., Ramos, P. M. & Serra, A. C. (2009) New Spectrum Leakage Correction Algorithm for Frequency Estimation of Power System Signals, *IEEE Trans. Instrum. Meas.*, Vol. 58, no. 5, pp.1670-1679

Schoukens, J., Pintelon, R. & Van hamme, H. (1992) The Interpolated Fast Fourier Transform: A Comparative Study, *IEEE Trans. Instrum. Meas.*, Vol. 41, No. 2, pp.226-232

Yao, Y. & Pandit, S.M. (1995) Cramér-Rao lower bounds for a damped sinusoidal process, *IEEE Trans. Signal Process.*, vol. 43, no. 4, pp. 878–885

Yoshida, I., Sugai, T., Tani, S., Motegi, M., Minamida, K. & Hayakawa, H. (1981) Automation of internal friction measurement apparatus of inverted torsion pendulum type, *J. Phys. E: Sci. Instrum.*, vol. 14, pp. 1201-1206

2

A Proposed Model-Based Adaptive System for DFT Coefficients Estimation Using SIMULINK

Omar Mustaf
Gulf University
Kingdom of Bahrain

1. Introduction

Spectral Analysis is of great importance in signal processing applications in general, and in the analysis and the performance evaluation of communications systems specifically. On the other hand the procedure of calculating/estimating the spectrum itself is also important, i.e. whether it's simple or complicated, especially when taking into account the limitations of onboard space in the parallel computation and VLSI implementation. Therefore, in response to the above, Widrow et al. proposed an adaptive method for estimating the frequency content of a signal through demonstrating a relationship between the Discrete Fourier Transform (DFT) and the Least Mean Square (LMS) algorithm, where the DFT coefficients are estimated by a new means using the LMS algorithm (Widrow et al., 1987). That was the original attempt of relating the DFT to the LMS adaptation rule. The main features of such a spectrum analysis are simplicity, adaptability, and suitability with parallel computations and VLSI implementation. This is owing to the nature of the LMS algorithm, which lends itself to this type of implementation.

Later on, Mccgee showed that Widrow's spectrum analyzer could be used as a recursive estimator for the sake of solving the exponentially-weighted least squares estimation and a filter bank model was deduced. The fundamental outcome of that work was that the LMS algorithm could act as a bank of filters with two modes of operation, which means when the LMS learning rate is chosen to be ½, the filter poles are located at the origin. This means equivalently that the LMS effective transfer functions are FIR filters; otherwise, the equivalent filter is IIR (Mccgee, 1989).

The Widrow's principal relation between the LMS and the DFT was then extended to the 2-Dimentional (2-D) case by Liu and Bruton, which directly resulted in the 2-D LMS spectrum analyzer. Here it was shown that the 2-D LMS algorithm has the advantage that allows concurrent computations of what was called an updating matrix and therefore the potential for very fast parallel computations of the 2-D DFT; however, there was no statement about how fast it was (Liu and Bruton, 1993).

Two years later, a generalization of the above mentioned 2-D LMS spectrum analyzer was demonstrated by (Ogunfunmi and Au, 1995). It was achieved by successfully extending the relation to other 2-D discrete orthogonal transforms. The simulations of that work showed the same results when imposing a frame of a test signal of size 32 by 32 to both a 2-D DFT

LMS based algorithm and a 2-D spectrum analyzer for the 2-D discrete cosine transform. The same results coincidence was demonstrated when comparing the estimated spectrum of the 2-D DFT LMS based with that of the 2-D discrete Hartly transform (DHT).

Beaufays and Widrow came back in 1995 to compare the LMS spectrum analyzer with the straightforward, non-adaptive implementation of the recursive DFT, and the robustness of the LMS spectrum analyzer to the propagation of round-off errors was demonstrated. Also, they showed that this property is not shared by other recursive DFT algorithms (Beaufays and Widrow, 1995).

In 1999 Alvarez et al. proposed the analog version of the LMS spectrum analyzer, where the coefficients are adapted independently by analog LMS structure which can be implemented in the VLSI technology. The main advantage gained from the work is that since the adaptation is carried out in the continuous time domain, real time computing speed was achieved (Alvarez et al., 1999).

This chapter aims mainly to propose a model-based simulation design procedure for realizing the relationship between the DFT and the LMS adaptation algorithm, using a very powerful simulation tool, namely SIMULINK, which runs under the MATLAB package. The proposed design is supposed to perform a spectral estimation using Widrow's adaptive LMS spectrum analyzer (Widrow et al., 1987). Therefore, this work attempts to provide a design procedure for that theoretical work. Specifically, this chapter proposes a model-based design procedure with simulation results to show the importance of this relation, which can be considered as an adaptive spectrum analyzer with wide range of applications in the design of modern digital communications transceivers and specifically the transform domain equalizers.

The simulation methodology is based on the SIMULINK environment and it adopts a block by block SIMULINK design procedure by which each *system unit* (system unit means, for example, phasor generator unit, LMS adaptation process unit) of the designed model is created at the block level and then assembled together properly to form the intended system model of Widrow's adaptive spectrum analyzer model.

After finalizing the design, test signals will be imposed to the model for verifying the validity of the DFT coefficients that are estimated adaptively using the LMS algorithm. The simulation results will be showed and discussed clearly in the simulation results discussion section. In addition, new frequency contents will be added to the test input signal during the running mode to test the property of adaptability to sudden changes in the input signal, thus seeing how the system will follow up these changes.

As a future work, a generalized SIMULINK model design procedure for any specified *n-point* spectral estimation is suggested as a further development of this work. Also, there will be a suggestion of how to use the proposed model in the design and simulation of frequency domain equalizers as an application.

Finally, the chapter includes a brief introduction to using SIMULINK, which is important, especially for those who do not have access to or have limited information about SIMULINK. It acts as a *clear, easy, short and concentrated guide* to understand and/or develop the work presented in this chapter.

2. The mathematical background of the Adaptive Spectrum Analyzer

In this section the mathematical model of the adaptive spectrum analyzer, which was proposed first by Widrow et al. (Widrow et al., 1987), is presented. The main theory and the useful result of that work is given in this section without going further into the detailed mathematical derivations and proof, as it is out of the scope of this chapter. The objective of the chapter is to use this demonstration to propose a SIMULINK model that is corresponding to the adaptive LMS spectrum analyzer. Also, the proposed simulation model might be adopted later as a basic building block in the design of adaptive two-layer structures for fast filtering as it will be illustrated in the future work section at the end of the chapter. The Widrow et al. model will be abbreviated as WASA throughout the rest of the chapter, which stands for Widrow Adaptive Spectrum Analyzer.

The key idea of WASA is that for an arbitrary signal to be analyzed in the frequency domain, it's possible to choose a number of phasors, according to a certain criterion, and then weight these phasors with adaptable weights that might be adapted by iterating the well-known LMS adaptation role. Upon choosing a suitable value for the adaptation rate, these weighted phasors are equal to the DFT coefficients of the signal under consideration. Fig. 1 shows the complete schematic diagram of the WASA model.

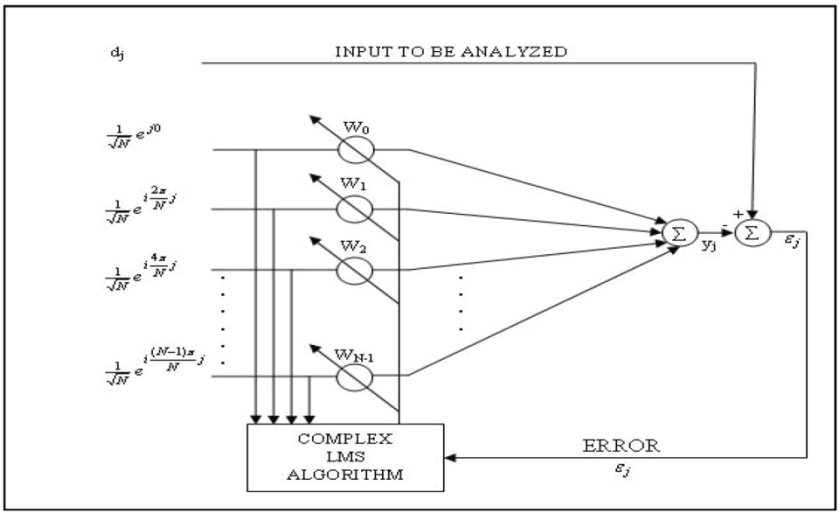


Fig. 1. Adaptive LMS spectrum Analyzer as proposed by (Widrow, et al., 1987)

The signal d_j is the input sampled version of the signal to be Fourier analyzed which is sampled at each time index j. The sampling time is T sec and the corresponding sampling frequency is $\Omega = {}^{2\pi}\!/_{T}$ rad/sec. It's the objective of WASA to resolve d_j to its corresponding DFT coefficients. The left hand side of Fig. 1 shows a set of complex exponentials, which represent the time domain phasors that are supposed to equal the DFT coefficients of d_j after weighting them with an adaptable weight vector. There are N phasors, where N refers to the desired number of the DFT coefficients. The corresponding frequencies of these set of phasors span the frequency range from DC up to the sampling frequency Ω, including the

fundamental frequency Ω/N. The fundamental phasor can be expressed in terms of the time index *j* as:

$$e^{i\frac{\Omega}{N}jT} \tag{1}$$

Where $i=\sqrt{-1}$. Now, since $\Omega.T = 2\pi$, then this phasor becomes:

$$e^{i\frac{2\pi}{N}j} \tag{2}$$

The rest of the other phasors are the power of equation (2).

In vector form these phasors might be written as:

$$X_j \cong \frac{1}{\sqrt{N}}\begin{bmatrix} 1 \\ e^{i\frac{2\pi}{N}j} \\ e^{i\frac{4\pi}{N}j} \\ . \\ . \\ e^{i\frac{2\pi(N-1)}{N}j} \end{bmatrix} \tag{3}$$

These phasors are to be weighted at each sampling time index *j* by the adaptable weight vector W_j, where

$$W_j \cong \begin{bmatrix} w_{j0} \\ w_{j1} \\ w_{j2} \\ . \\ . \\ w_{jN-1} \end{bmatrix} \tag{4}$$

The factor $\frac{1}{\sqrt{N}}$ is a normalization factor and it's used to simplify the analysis of the system of Fig. 1, because it pulls the power of the phasors' vector, $\overline{X}_j$ to unity, as it is shown in the main literature of (Widrow, et al, 1987). The weight vector components w_{j0}, w_{j1},…., w_{jN-1} are updated at each sampling time index j in accordance to the well-known LMS adaptation rule,

$$W_{j+1} = W_j + 2\mu\varepsilon_j\overline{X}_j \tag{5}$$

Where

$$\varepsilon_j = d_j - y_j \tag{6}$$

is the instantaneous error between the input signal d_j and the spectrum analyzer output, y_j which is given by:

$$y_j = X_j{}^T W_j \tag{7}$$

whereas, μ represents the LMS adaptation rate (speed) and $\overline{Xj}$ is the complex conjugate of X_j. (Widrow, et al., 1987) concluded that when the learning speed is set to 0.5, then the output of the spectrum analyzer of Fig.1 is equal to the DFT of the input signal d_{j-1}. Referring to Fig. 1, the spectrum analyzer output is:

$$\begin{pmatrix} LMS \\ spectrum \\ analyzer \\ output \\ vector \end{pmatrix} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & & & & \\ & e^{i\frac{2\pi}{N}j} & & & \\ & & e^{i\frac{2\pi(2)}{N}j} & & \\ & & & \ddots & \\ & & & & e^{i\frac{2\pi(N-1)}{N}j} \end{bmatrix} Wj \tag{8}$$

$$\therefore DFT[d_{j-1}] = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & & & & \\ & e^{i\frac{2\pi}{N}j} & & & \\ & & e^{i\frac{2\pi(2)}{N}j} & & \\ & & & \ddots & \\ & & & & e^{i\frac{2\pi(N-1)}{N}j} \end{bmatrix} Wj \tag{9}$$

This conclusion is demonstrated through the proposed SIMULINK model of this chapter through considering real time signals for the input d_j, and then it will be compared with that of the standard sliding DFT system in the simulation result section.

3. Guide for using SIMULINK

This is a brief guide to using SIMULINK regarding to the modelling and simulation of dynamic systems. It gives a general introduction to SIMULINK, its importance and some useful tips of modelling with SIMULINK. The guide provides sufficient knowledge to understand the proposed model of this chapter, especially for those who have a limited knowledge of SIMULIMK. A simple DSP example, namely a 'tapped delay line' of an arbitrary signal, is adopted in the end of the guide as a simple but a good practice of modelling with SIMULINK.

Finally, it is important here to indicate that the entire guide, and even the design of the proposed system of this chapter, assumes that the MATLAB is installed on a platform with a

Microsoft Windows operating system and not a Macintosh environment. Consequently, this may lead to some differences to those who are using a Macintosh operating system when dealing with the graphical user interfaces.

3.1 SIMULINK at a glance

SIMULINK which stands for SIMUlation and LINK, is a simulation tool that runs under the MATLAB package which stands for MATrices LABoratory, the main software by Mathworks. SIMULINK combines both the simplicity and the high capability of simulation and modelling of dynamic and embedded systems. SIMULINK can be used efficiently in the design, implementation, simulation and testing of a wide range of applications like communications, controls, signal processing, video processing and image processing. Although SIMULINK requires the installation of MATLAB on the platform of simulation, in general it doesn't require prior knowledge of programming using MATLAB, which is the case here in our proposed model-based system design; however, in other cases it does require such knowledge, but it is out of the scope of this chapter to go further in explaining such cases.

SIMULINK, as stated earlier, works under the MATLAB package, so it requires that MATLAB is installed on the simulation platform, normally a personal computer (pc), and the installation of SIMULINK choice should be selected during the installation process. Thus, the first step is running MATLAB. MATLAB can be run by double-clicking on the MATLAB icon on the desktop, and then SIMULINK is easily opened by either clicking on the SIMULINK icon in the MATLAB toolbar or by writing SIMULINK in what is called the *command window*.

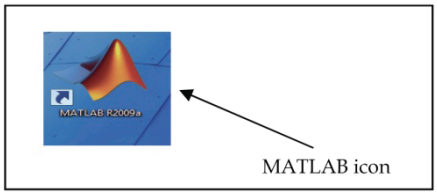


Fig. 2. Opening a MATLAB session

SIMULINK can be opened in different ways. The simplest way is by clicking on the SIMULINK icon in the MATLAB toolbar as illustrated in Fig. 3.

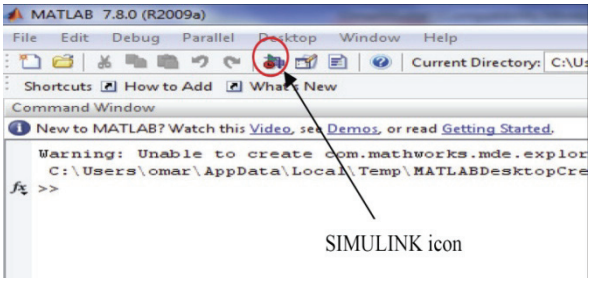


Fig. 3. Opening SIMULINK

Fig. 4 shows the SIMULINK Library Browser window that appears after clicking on the SIMULINK icon. This is where you can find all the components that you may add to your model. Components are known as blocks in the SIMULINK context. Once the SIMULINK library browser is opened, a modeling environment has to be opened. This is called a model file, which is opened from the SIMULINK library browser toolbar by clicking on the new model icon as illustrated in Fig. 4. Fig. 5 shows a blank new model file.

Fig. 4. The SIMULINK Library Browser

In this model file you can create, connect all of your system components, simulate, change simulation parameters and finally view and/or print the results. The model file has to be saved with a suitable name and it will be saved in a default folder, namely MATLAB, which is created automatically during the installation period. Fig. 6 clarifies how to save a SIMULINK model.

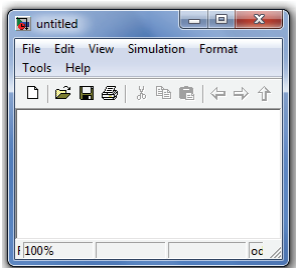


Fig. 5. New SIMULINK model file

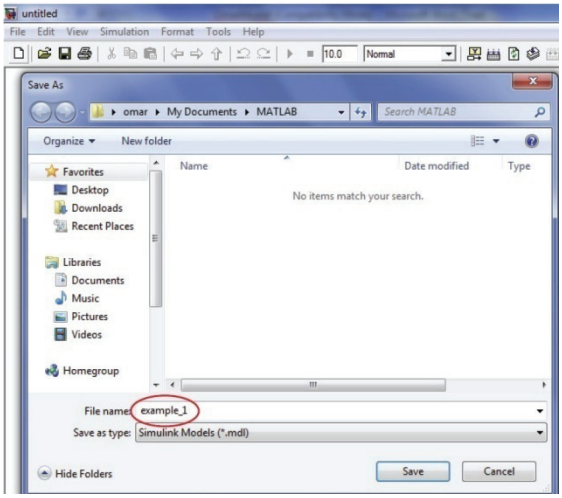


Fig. 6. A standard Windows operation to save SIMULINK model files

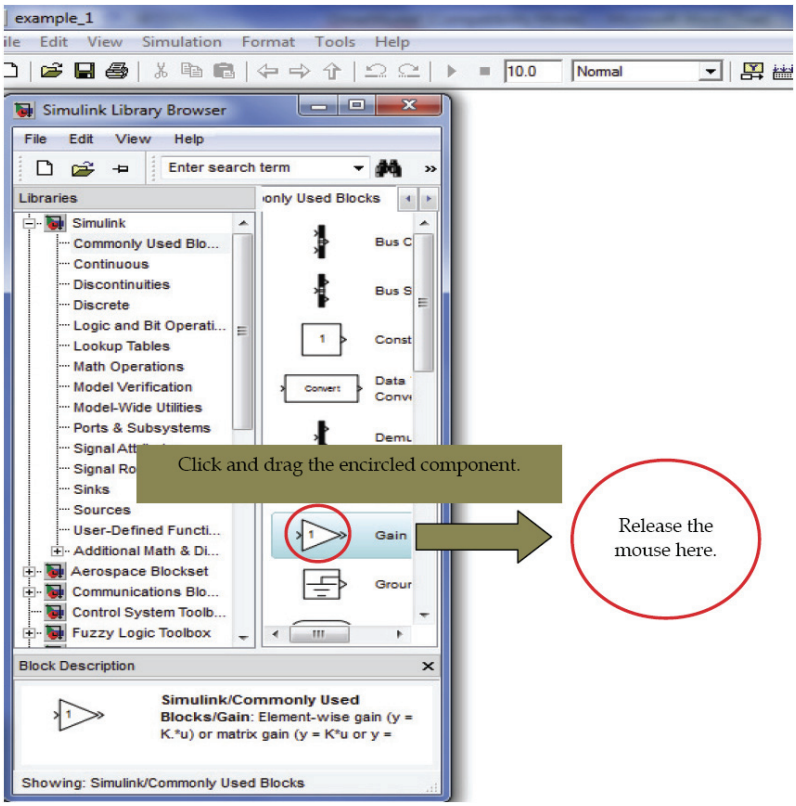


Fig. 7. Inserting blocks in a SIMULINK model

Now, the entire components/blocks that you may wish to add to your model are available at the SIMULINK LIBRARY BROWSER. The next step is inserting and connecting the blocks of the model. A block can be inserted from the library browser very easily by left-clicking on the block, holding down the button, dragging it and then releasing the mouse in the model file. The procedure of adding a block to a model is illustrated in Fig. 7. It is useful here to indicate that a brief explanation about the mathematical operation of the selected block appears at the bottom of the library browser, as is illustrated in Fig. 7.

Similarly, the entire model's blocks might be inserted in the same way. At this point, after inserting all of the required model components, you may rearrange the locations of the components within the same model file in such a way that best matches the schematic diagram of the simulated system. The next important step is linking the blocks of the model. Fig. 8 shows how to link two blocks together automatically, while Fig. 9 shows how to make a branch line.

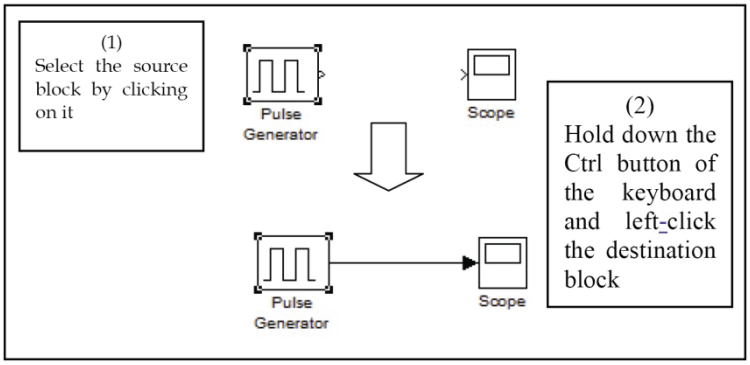


Fig. 8. Linking blocks in the model file

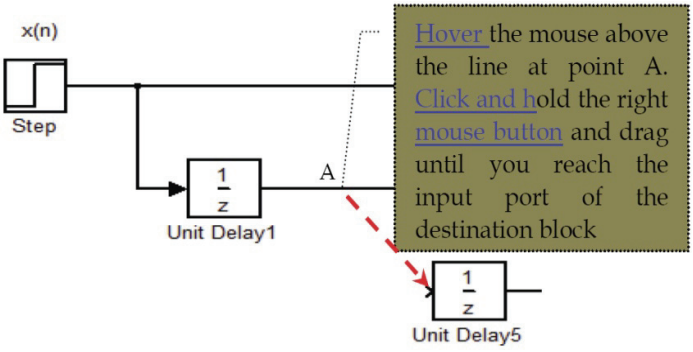


Fig. 9. Making a branch line

Each SIMULINK block has its own simulation parameters, and these parameters can be reached by double-clicking on the block and then it can be changed according to the simulation requirements. As an example, Fig. 10 shows how to change the gain value of a gain block.

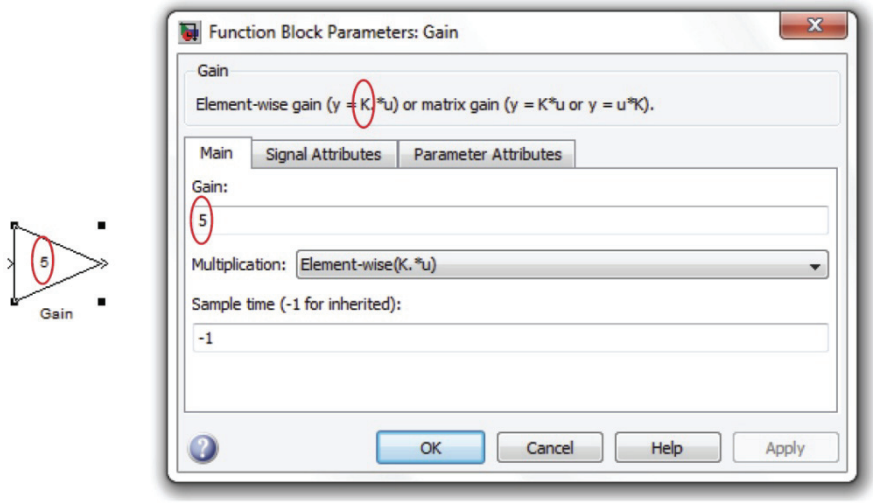


Fig. 10. SIMULINK block parameter viewing/changing

Finally, the facility of viewing the simulation results in various ways, adds more flexibility in performing simulations using SIMULINK. Actually, there are several devices that you might use for this purpose, namely *scope*, *x-y plotter*, *display*, *spectrum analyzer*, and *to work space*. The illustrative example in the next subsection shows the use of a *scope* device to display the simulation results. The user has the choice of inserting the scope device anywhere in the model to view the desired signal in the time domain.

3.2 An illustrative example of running a SIMULINK model: A tapped delay line model

This example shows a very common process in the DSP applications, which is the creation of a tapped delay line, which is a simple practice to learn how to create/run SIMULINK models. So, the idea is to create a lines that carry delayed versions of an arbitrary signal, say x(n). Suppose that there is a need to create delayed versions of x(n) of up to x(n-4). The model will be created on the last saved model, i.e. the one shown in Fig. 6 which holds the model name 'example_1'.

The model creation and simulation may be summarized in the following steps.

Step 1. Insert four unit delays from the discrete blocks category.
Step 2. Insert an arbitrary input signal from the source blocks category, and let it be a step block.
Step 3. Insert a scope block from the sinks category of blocks. Change the number of scope inputs to five, by double-clicking on the scope block and then changing the number of the axes parameter as shown in Fig. 11.
Step 4. Re arrange all of the blocks inside the model and connect them. Follow the procedure given in Fig. 9 to connect a line with a block.
Step 5. Click on the run button, as indicated in Fig. 12, to run the model for a default simulation time of 10 sec.

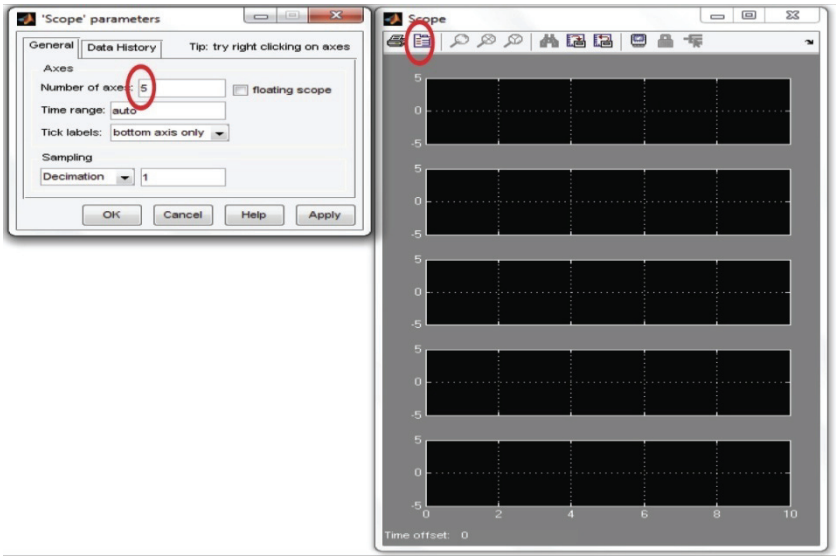


Fig. 11. Changing the number of inputs of the scope block

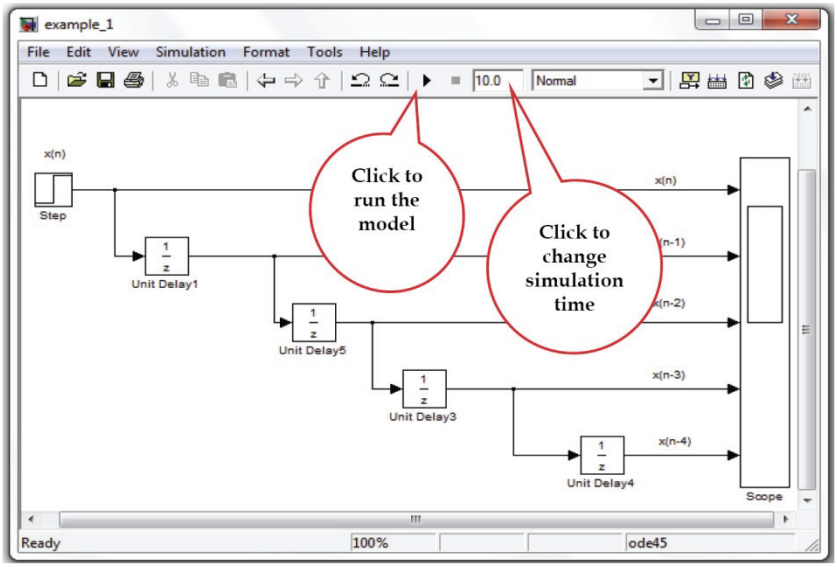


Fig. 12. Tapped delay line model

SIMULINK provides the facility of creating subsystems. This means that you have the ability to create your own blocks by combining a number of blocks together to perform a certain function. It is useful here to explain how to create a subsystem using the following example, as this facility will be adopted in this chapter in the creation of the LMS spectrum analyzer model.

Suppose you need to use a tapped delay line of certain delays many times in your model or perhaps you want to create your own library of blocks within your SIMULINK package. SIMULINK provides the ability to do that by grouping them within one block with a default name of *subsystem 1*. This can be realized by selecting all the corresponding components and then right-clicking on one of them and choosing the create subsystem option form the menu as illustrated in Fig. 13.

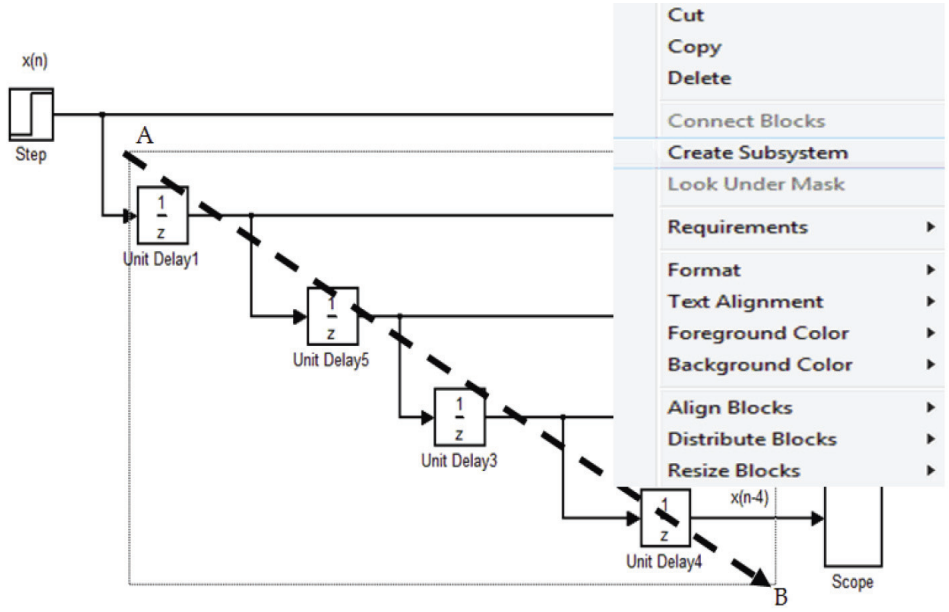


Fig. 13. Subsystem creation; one approach

Now your subsystem has been created, as shown in Fig. 13, and it is ready to use. You can resize your subsystem in order to make it fits its location in the model and also rename the input and/or the output ports in order to make your subsystem more organized and more expressive. Some of the useful operations on a subsystem are shown in Fig. 14.

4. The proposed WASA SIMULINK based design

In this section, a SIMULINK model which corresponds to the theoretical model of WASA is created and later it will be tested for a 4-points DFT case. The complete model is shown in Fig. 15. This model consists of two main subsystems, namely the phasor creation subsystem and the LMS adaptation subsystem. Other blocks are the summer, subtractor, and the mixers bank. The phasor creation subsystem is responsible for the generation of the frequency vector X_j which is defined by equation 3. The procedure that is followed here to realize each phasor is to combine two sine and cosine source signals to form the required phasor according to Euler's formula;

$$e^{\pm j\theta} = cos(\theta) \pm j\, sin(\theta) \tag{10}$$

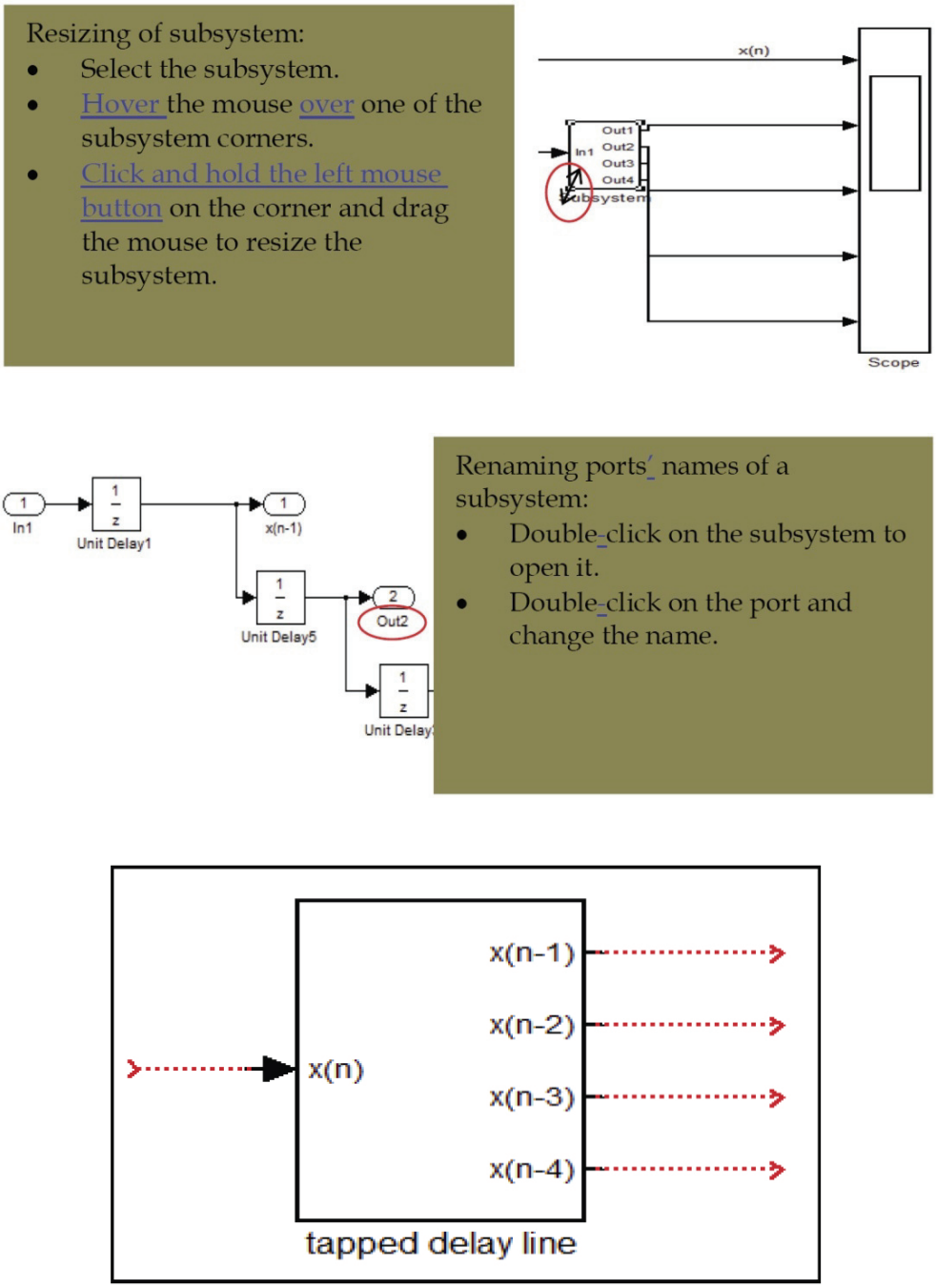


Fig. 14. Useful operations on a subsystem

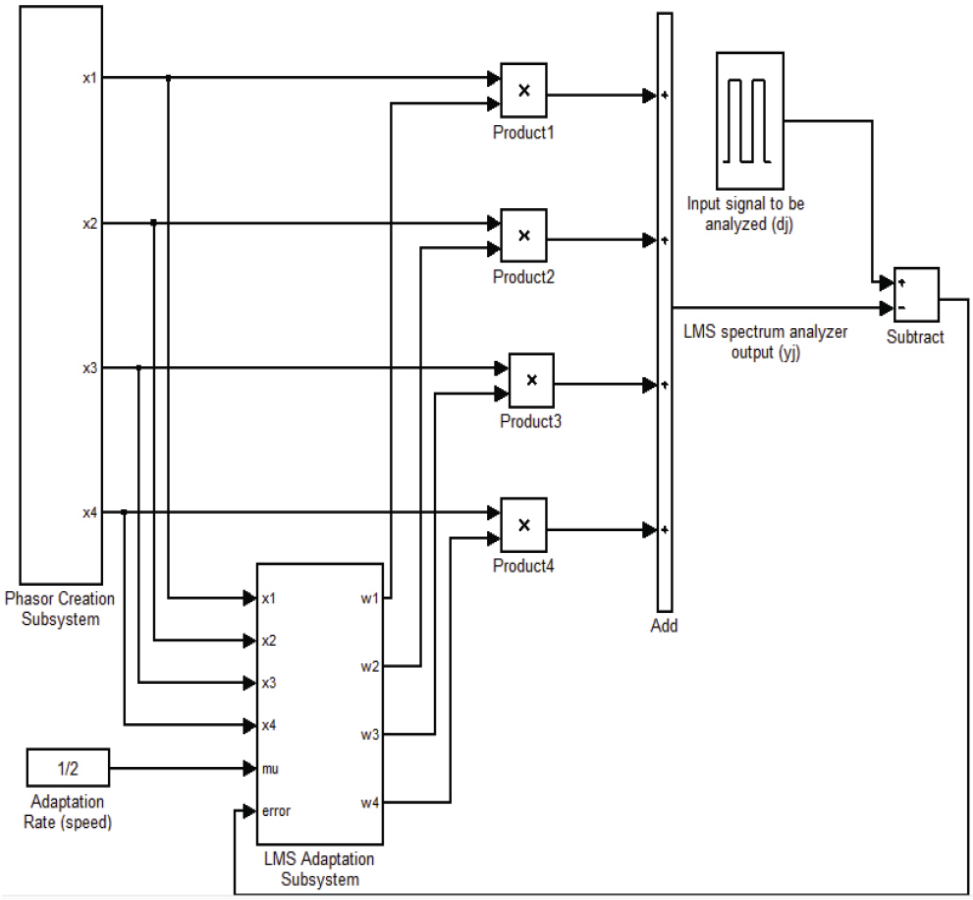


Fig. 15. The proposed WASA SIMULINK model

Fig. 16 shows a proposed 4points DFT phasor creation subsystem. Coming down to the block level, each phasor is composed of sine and cosine signals with the corresponding frequency band that starts from the DC component and ends with the $(\frac{2\pi(N-1)}{N})$ component.

The outputs from this subsystem are then fed to two places: firstly, to the mixer bank for the sake of calculating the spectrum analyzer output, y_j; secondly, to the second main subsystem. This is the LMS adaptation process because these phasors are included in the adaptation process of the weight vector W_j. Fig. 17 shows the contents of the LMS adaptation subsystem. The blocks inside the red rectangle, which are the simulation of equation 5, perform the adaptation role for each weight of the weight vector. Also, this figure shows how the instantaneous error and the adaptation rate constant are fed to all of the adaptation roles of the weights, w_j.

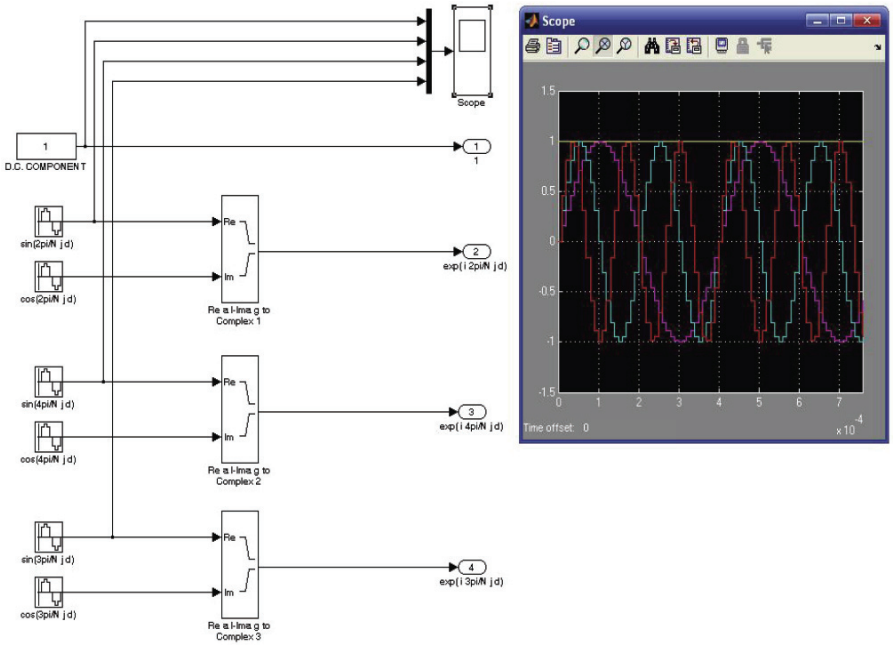


Fig. 16. The block connection details inside the phasor creation subsystem

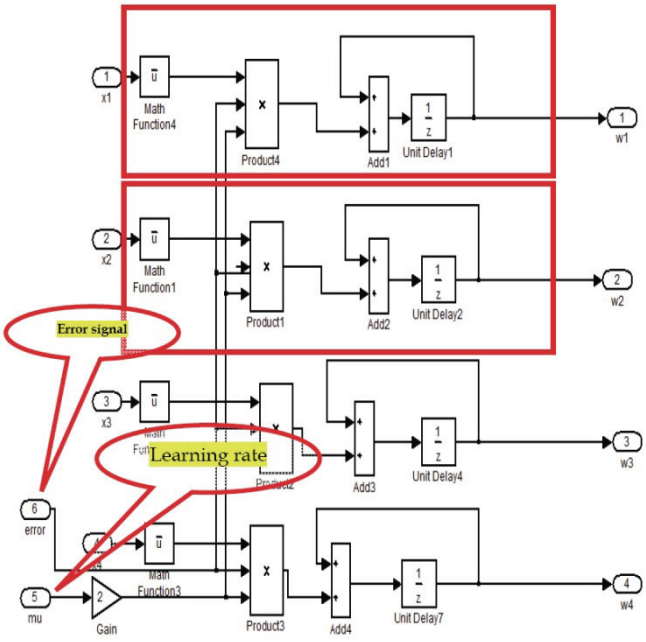


Fig. 17. The detailed blocks inside the LMS adaptation subsystem

The mechanism of operation of this model might be summarized as follows. The phasors are created in the phasor creation subsystem at each time sample index j. These vectors are weighted by the pre-loaded values of the weight vector through the mixer bank, then the weighted phasors are summed together to generate a sample of the LMS spectrum analyzer output y_j. The latter is subtracted from the input signal sample d_j to generate the instantaneous error signal ε_j. This error is fed to the LMS adaptation subsystem to update each weight gain of the weight vector instantaneously with the adaptation speed μ and the scaled phasor vector X_j. Now this model is finalized, and it is ready for setting the simulation parameters and exploring the results.

5. Simulation and results discussions

In this section, the performance of the proposed WASA SIMULINK model is tested using the following two scenarios.

1. Imposing a test signal and the results, i.e. the estimated DFT coefficients of the input test signal (d_j), are compared with those which are obtained from the steady flow DFT operation.
2. Imposing a test input signal and while running the model. New frequency contents are introduced in this test signal to test the adaptability feature of the proposed model and snapshots are taken to show how the power spectrum is updated to include the newly-added frequency contents.

The simulation parameters, results and results discussions for the two scenarios are as demonstrated below:

Scenario (1)

Simulation parameters

Simulation time (run time): 10 sec.

WASA output size: 4-DFT coefficients.

Input signal, d_j: pulse signal with a period time of 4μsec, 50% duty cycle, amplitude of 1volts, and sampling rate of 1 MSps.

Simulation results and discussion

The simulation results for this scenario are compared with those obtained from the conventional steady flow DFT. The steady flow DFT operation is designed from the tapped delay line example which was realized previously in Fig. 12 and a DFT SIMULINK block. The results for the proposed WASA model and the steady flow DFT are shown in Fig. 18.

The *display* block is adopted to display the calculated and the estimated DFT coefficients from the sliding DFT and the proposed WASA models respectively. After performing the running of the model, it is clearly seen that the results are absolutely coincident as they appear in their respective display blocks.

Another interesting result to be shown here is the instantaneous error signal, ε_j. Fig. 19 shows various cases for the sampled version of this signal after running the proposed model three times for three values of the learning speed μ. The results justify the mathematical proof in (Widrow, et al, 1987) which is given by equation (9) and states that if μ is chosen to be 0.5, the output of the adaptive LMS spectrum analyzer will be equal to the DFT of the input sampled signal at the previous time instance. Otherwise, ε_j will never reach zero as the case shows in Fig. 19 (a). Fig. 19 (c) shows that when μ is chosen to be more than 1, ε_j is increasing up to infinity as the LMS algorithm becomes unstable.

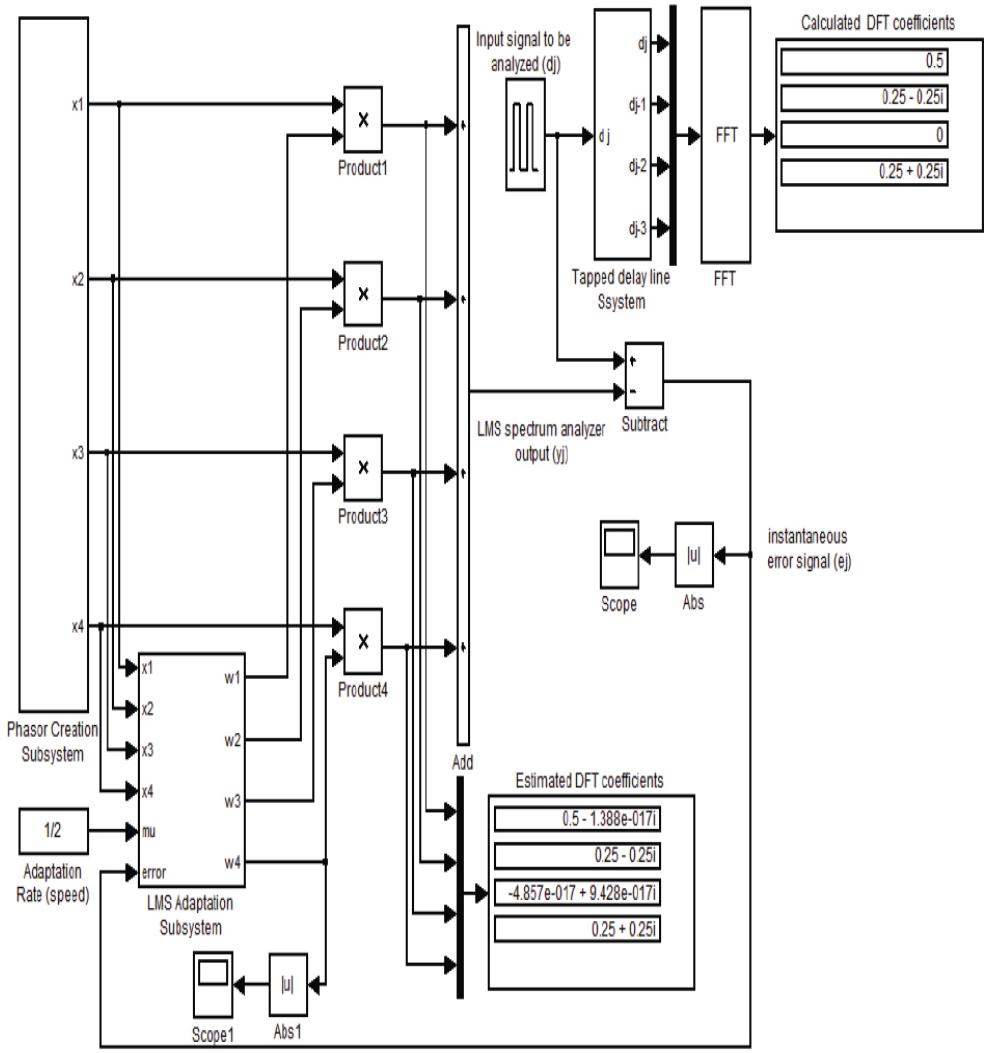
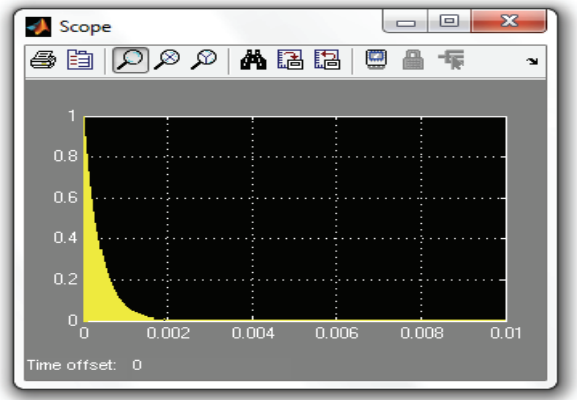
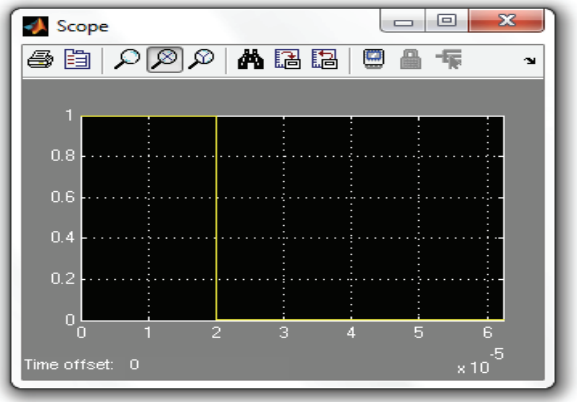


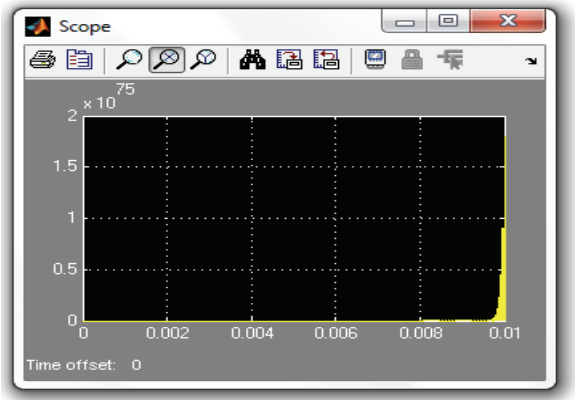
Fig. 18. 4 points DFT WASA model results as compared with sliding DFT



(a) μ = 0.05



(b) μ = 0.5



(c) μ = 1.5

Fig. 19. The effect of the learning speed, μ, on the instantaneous error signal, ε_j

Scenario (2)

Simulation parameters

Simulation time: 0.1 sec.

WASA size: 4-DFT points.

Input test signal: $d(t) = sin(2\pi f_1 t) + sin(2\pi f_2 t).u(t - t_o)$

f_1= 100 KHz.
f_2= 200 KHz.
t_o= 0.01 sec.

Amplitude of 1volts, and sampling rate of 1MHz.

Simulation results

The second test is the adaptability test. This is carried out by imposing a composite signal of two sinusoidal signals to the model to serve as d_j.

Simulation results and discussion

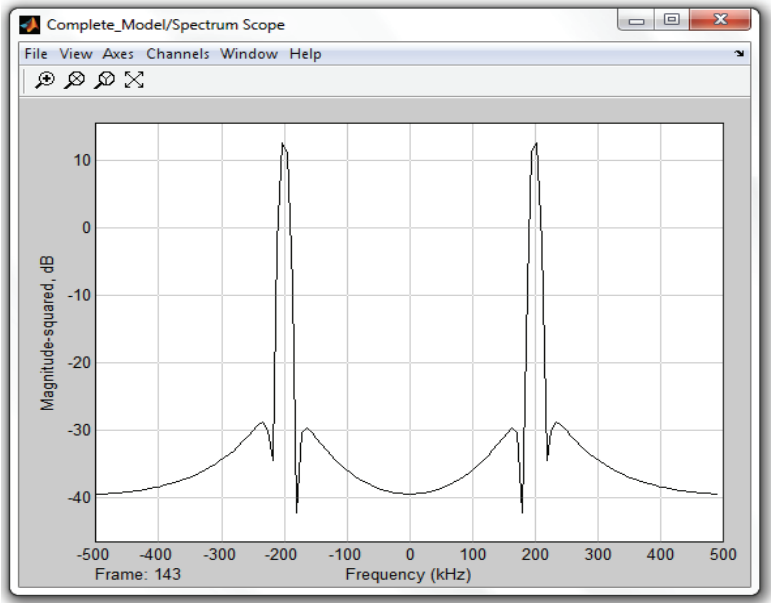
It is clear that this signal contains two fundamental frequencies, f_1 and f_2, but the f_2 component does not appear in the spectrum of the signal until the time instance t_o as the second term of $d(t)$ is multiplied by the shifted unit step function. Therefore, the WASA proposed SIMULINK model operates on the term $sin(2\pi f_1 t)$ to estimate the f_1 component until the time instance t_o, then the weight vector will be adapted by the LMS role again to estimate the frequency content of the second term, which is f_2. The results are viewed using the Spectrum scope block which is connected at the output node of the WASA model.

Two snap shots are taken from the spectrum scope block: one before t_o and the other after t_o, as shown in Fig. 20.

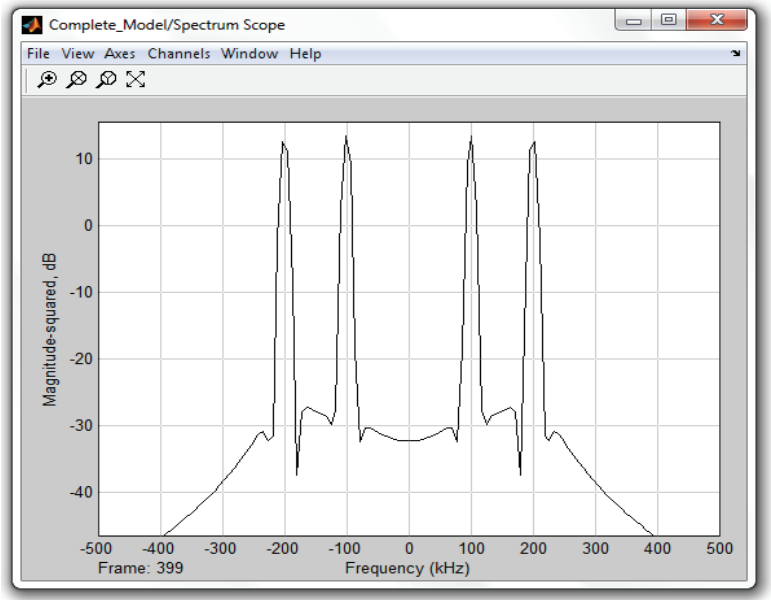
Finally, it is clearly seen that the proposed model simulated the WASA system perfectly; however, there is an issue that seems to be a drawback for the procedure of the proposed WASA model creation. The point is that when the size of the system is small or moderate, i. e. the size of the DFT operation, the system creation is quite simple and doesn't require a lot of effort to finalize it, which is the case for points 2 to 16. But what about the points 32 and up; 128, for example? The issue is that the system will require a bulk of blocks to be embedded in, and the branching of lines across the model will be tedious work. So, there must be another strategy to be adopted in the model creation phase. This will be clarified in the suggestions of future work section.

6. Future work

Actually, there are two tracks to be followed for the use of the proposed WASA SIMULINK-based design and/or developing the proposed adaptive spectrum analyzer. First is to generalize the model by generalizing the two main subsystems of the model, the phasor creation block and the LMS adaptation, to work with any specified number of DFT points. This would be realized by using MATLAB programming to create two script files that represent the mathematical relations of these two subsystems, then turning these scripts into



(a)



(b)

Fig. 20. Results for the second test/adaptability test. (a) Frequency spectrum before adding the 10KHz component (b) Spectrum after adding the 10KHz component

SIMULINK blocks and embedding them later with the main model. This is a facility that SIMULINK provides, which makes the work look more professional and does not require huge and tedious wiring work during the model creation. The input parameter of the phasor creation block, for example, would be the number of the preferred DFT coefficient and the sampling rate.

Secondly is to extend the work of this chapter in the design of a class of digital modulation receivers. This could be the design of frequency domain adaptive equalizers, which adopt the frequency domain of the signals to remove the transmission channel noise from the received contaminated signals. Therefore, one can adopt the proposed WASA model in the design of the two layer linear structure for fast adaptive filtering, which was presented by (Beaufays, Widrow, 1995). Fig. 21 shows WASA location in the system inside the thick box.

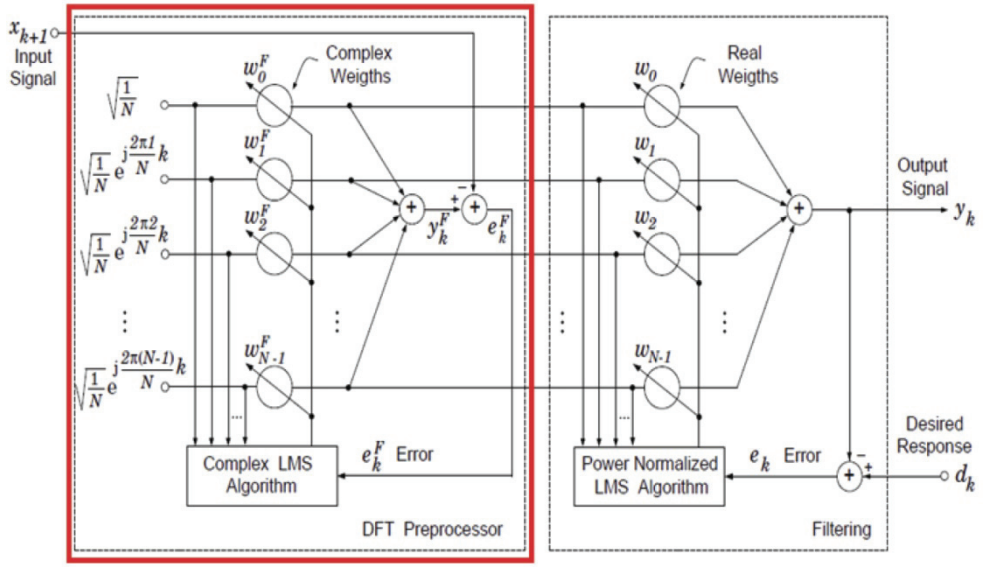


Fig. 21. WASA suggestion for adaptive frequency equalization (Beaufays, Widrow, 1995)

7. Conclusion

A model-based design is proposed to simulate the LMS adaptive spectrum analyzer. A SIMULINK simulation environment is used for its simplicity and high capabilities. The proposed model successfully simulated Widrow's model for 4-DFT points. The results shows a coincidence between the estimated DFT coefficients from the proposed model and the results calculated from a standard steady flow DFT model. Also, it justifies that when the adaptation rate, μ of the model is set to 0.5, then a set of weighted phasors exactly corresponds to the DFT coefficients of the applied input to the model. In addition, new frequency contents are added to the test input signal during the run (online) as a separate test to assess the adaptability property of the spectrum analyser model for the sudden changes in frequency content of the input signal. As a future work, a generalized SIMULINK model is suggested for any specified n-point DFT.

8. References

Alvarez, A. O., and Rivera, L. N., Meana, H. P. (1999). Real time high frequency spectrum analyzer. *Proceedings of 42nd symposium on circuits and systems*, Vol. 2, (August 1999), pp. (753-756)

Beaufays F., and Widrow, B. (1994). Two-Layer Linear Structure for Fast Adaptive Filtering. Proceedings of WCNN, San Diego, USA, Vol. 3, June 1994

Beaufays, F., Widrow, B. On the advantages of the LMS spectrum analyzer over nonadaptive implementations of the sliding –DFT. (1995). *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, Vol. 42, No. 4, (April 1995), pp. (218-210)

Liu, B., Bruton, L. T. (1993). The two-dimentional complex LMS algorithm applied to the 2-dD DFT. *IEEE Transactions on Circuits and Systems-II: Analog and digital signal processing*, Vol. 40, No. 5, (May 1993), pp. (337-341)

Macgge, W. F. Fundamental relationship between LMS spectrum analyzer and recursive least squares estimation. (1989). IEEE Transactions on Circuits and Systems. Vol. 36, No. 1, (January 1989), pp. (151-153)

Ogunfunmi, T., Au, M. (1994). 2-D discrete orthogonal transforms by means of 2-D LMS adaptive algorithms, Poceedings of signals, systems and computers, 0-8186-6405-3, Pacific Grove , CA, USA, November 1994

Widrow, B., Baudrenghien, P., Vetterli, M., and Titchener, P. (1987). Fundamental Relations Between the LMS Algorithm and the DFT. IEEE Transactions on Circuits and Systems, Vol. 34, No. (7), (July 1987), pp. (814-820)

3

FFT-Based Efficient Algorithms for Time Delay Estimation

Renbiao Wu, Wenyi Wang and Qiongqiong Jia
Tianjin Key Lab for Advanced Signal Processing
Civil Aviation University of China, Tianjin
P.R.China

1. Introduction

Suppose that we have a single sensor receiving a superimposition of attenuated and delayed replicas of a known signal plus noise. From the received data we want to estimate the arrival times of the various replicas and their (complex or real) attenuation coefficients (gains). This is the well-known time delay estimation problem which occurs in many fields including radar, active sonar, wireless communications, Global Navigation Satellite System (GNSS), nondestructive testing, geophysical/seismic exploration, and medical imaging. In this chapter, we will focus on time delay estimation based on one sensor with known probing signal shapes.

The most well-known time delay estimator is the matched filter approach. If there is only one signal or the overlapped signals are separated in time by an interval that is much greater than the width of the signal autocorrelation function, then the matched filter is the optimal estimator when the noise is white Gaussian (Ehrenberg et al., 1978) . The resolution capability of the matched filter approach depends on the signal bandwidth and the larger the signal bandwidth, the better the resolution. However, in many situations there exist some practical limitations on increasing the bandwidth of the transmitted signals. How to resolve closely spaced overlapping noisy echoes has attracted the attention of researchers from many fields for several decades.

Several approaches have been suggested for this problem and many of them benefit from the recent development of high resolution sinusoidal frequency estimation and Direction of Arrival (DOA) estimation techniques. Sinusoidal frequency estimation techniques such as MUSIC (Schmidt, 1986), Linear Prediction (Tufts et al., 1982), and MODE (Stoica & Nehorai, 1990) are applied to the time delay estimation problem in (Bian & Last, 1997; Kirsteins, 1987; Kirsteins & Kot, 1990). However, these approaches are only applicable to signals with flat (rectangular) band-limited spectra. Several Maximum Likelihood (ML) approaches have also been suggested for this problem. Multidimensional global optimization algorithms are presented in (Bell & Ewart, 1986; Blackowiak & Rajan, 1995; Manickam et al., 1994) to analyze a special class of ocean acoustic data that has very oscillatory autocorrelation functions. An efficient approach based on the Expectation Maximization (EM) algorithm (Moon, 1996) is proposed in (Feder & Weinstein, 1988) that decouples the complicated multidimensional optimization problem into a sequence of multiple separate one-dimensional optimization problems. However, its convergence depends highly on the initialization method used and

no systematic initialization method is given in (Feder & Weinstein, 1988). More recently, time delay estimation lower bounds (Liu et al., 2010) and time delay estimation for small-samples (Gedalyahu & Eldar, 2010) are also studied.

In this chapter, a family of relaxation- and FFT-based new algorithms was proposed for different scenarios. The remainder of this chapter is organized as follows. In Section 2, we formulate the problem of interest. Section 3 describes the new algorithms. In Section 4, some applications utilizing the new algorithms are provided. Section 5 concludes the whole chapter.

2. Data model and problem formulation

Time delay estimation is a well-known traditional problem occurring frequently in radar, active sonar, and many other fields. In this problem, the waveform received at a single sensor consists of delayed replicas of the transmitted signal with different gains. The gains reflect the scattering property of the targets or multipath channel transmission features. The received signal waveform $y(t)$ can be described as

$$y(t) = \sum_{l=1}^{L} \alpha_l s(t - \tau_l) + e(t) \,, \quad 0 \le t \le T, \tag{1}$$

where $s(t), 0 \le t \le T_0$, represents the known transmitted signal (complex or real valued), $y(t)$ denotes the received signal, which is composed of L replicas of $s(t)$ with different (complex or real valued) gains $\{\alpha_l\}_{l=1}^{L}$ and real valued delays $\{\tau_l\}_{l=1}^{L}$, and $e(t)$ is the receiver noise, which is modeled as a zero-mean Gaussian random process.

Usually, the above received analog signal is sampled for digital signal processing. To avoid aliasing, we must sample $y(t)$ according to the bandwidth of $s(t)$. Let B_s denote the double-sided bandwidth of $s(t)$. Then $y(t)$ must be sampled with the sampling frequency f_s satisfying

$$f_s \ge B_s \,. \tag{2}$$

After A/D conversion, the sampled received signal has the form

$$y(nT_s) = \sum_{l=1}^{L} \alpha_l s(nT_s - \tau_l) + e(nT_s), \quad n = 0, 1, \cdots, N-1, \tag{3}$$

where T_s is the sampling period and is equal to the reciprocal of the sampling frequency f_s.

Our problem of interest herein is to estimate $\{\alpha_l, \tau_l\}_{l=1}^{L}$ from $\{y(nT_s)\}_{n=0}^{N-1}$ with known $s(t)$, $0 \le t \le T_0$, or $\{s(nT_s)\}_{n=0}^{N-1}$.

Although we could solve the estimation problem in the time-domain (Bell & Ewart, 1986; Blackowiak & Rajan, 1995; Bruckstein et al., 1985; Feder & Weinstein, 1988), we shall consider below solving the problem in the frequency domain and propose a family of relaxation-based algorithms for different scenarios.

Let $Y(k)$, $S(k)$, and $E(k)$, $k = -N/2, -N/2+1, ..., N/2-1$, denote the discrete Fourier transforms (DFT's) of $y(nT_s)$, $s(nT_s)$, and $e(nT_s)$, respectively. Provided that aliasing is negligible, then $Y(k)$ can be written as :

$$Y(k) = S(k) \sum_{l=1}^{L} \alpha_l e^{j\omega_l k} + E(k), \tag{4}$$

where

$$\omega_l = -\frac{2\pi\tau_l}{NT_s}\,. \tag{5}$$

Note that the time delay estimation problem is similar to the sinusoidal parameter estimation problem except that the exponential signals are weighted by the known signal spectrum. If we divided both sides of (4) by $S(k)$, the problem would become identical to the sinusoidal parameter estimation problem. Yet we should not do so for the following reasons: first, $S(k)$ could be zero for some k; second, the noise $E(k)/S(k)$ will no longer be a white noise even when $E(k)$ is white; third, when $E(k)$ is a white noise, the larger the $S(k)$ at sample k, the higher the signal-to-noise ratio (SNR) of the corresponding $Y(k)$ and hence dividing $Y(k)$ by $S(k)$ will de-emphasize those $Y(k)'s$ that have high SNRs. Because of this, many well-known sinusoidal parameter estimation algorithms, such as MUSIC (Schmidt, 1986), ESPRIT (Roy et al., 1986), PRONY (Kay, 1988), are not directly applicable to our problem of interest. Using MODE (Stoica & Nehorai, 1990) would require a multidimensional search over a parameter space because we can no longer reparameterize the MODE cost function via the coefficients of a polynomial.

3. FFT-based new algorithms

3.1 *W*eighted Fourier transform and *RELAX*ation (WRELAX) algorithm

We consider below estimating the unknown parameters by minimizing the following nonlinear least squares (NLS) criterion:

$$C_1(\{\alpha_l, \omega_l\}_{l=1}^{L}) = \sum_{k=-N/2}^{N/2-1} \left| Y(k) - S(k) \sum_{l=1}^{L} \alpha_l e^{j\omega_l k} \right|^2 . \tag{6}$$

When $e(nT_s)$ is a zero-mean white Gaussian random process, $E(k)$ is also white since DFT is a unitary transformation. For this white noise case, the NLS approach is the same as the ML method. When $E(k)$ is not white, however, the NLS approach is no longer the ML method. However, it has been shown in (Li & Stoica, 1996) that the NLS approach can still have excellent statistical accuracy.

Minimizing $C_1(\{\alpha_l, \omega_l\}_{l=1}^{L})$ with respect to the unknown parameters is a highly nonlinear optimization problem. The cost function has a complicated multimodal shape with a very small attraction domain, which makes it very difficult to find the global minimum. Below, we present a relaxation based optimization algorithm to obtain the NLS parameter estimates. Before we present our approach, let us consider the following preparations. Let

$$\mathbf{Y} = \left[\, Y(-N/2)\; Y(-N/2+1) \cdots Y(N/2-1) \,\right]^T, \tag{7}$$

$$\mathbf{S} = \mathrm{diag}\left\{\, S(-N/2)\; S(-N/2+1) \cdots S(N/2-1) \,\right\}, \tag{8}$$

$$\mathbf{E} = \left[\, E(-N/2)\; E(-N/2+1) \cdots E(N/2-1) \,\right]^T, \tag{9}$$

and

$$\mathbf{a}(\omega_l) = \left[e^{j\omega_l(-N/2)} \; e^{j\omega_l(-N/2+1)} \; \cdots \; e^{j\omega_l(N/2-1)} \right]^T , \tag{10}$$

where $(\cdot)^T$ denotes the transpose. Denote

$$\mathbf{Y}_l = \mathbf{Y} - \sum_{i=1, i\neq l}^{L} \hat{\alpha}_i [\mathbf{S}\mathbf{a}(\hat{\omega}_i)] \tag{11}$$

where $\{\hat{\alpha}_i, \hat{\omega}_i\}_{i=1, i\neq l}^{L}$ are assumed to be given. Consider first the case where $\{\alpha_l\}_{l=1}^{L}$ are complex valued. Let

$$\mathbf{b}(\omega_l) = \mathbf{S}\mathbf{a}(\omega_l), \quad l = 1, 2, \cdots, L. \tag{12}$$

Then (6) becomes

$$C_2(\alpha_l, \omega_l) = \| \mathbf{Y}_l - \alpha_l \mathbf{b}(\omega_l) \|^2, \tag{13}$$

where $\| \cdot \|$ denotes the Euclidean norm. Minimizing $C_2(\alpha_l, \omega_l)$ with respect to α_l yields the estimate $\hat{\alpha}_l$ of α_l

$$\hat{\alpha}_l = \frac{\mathbf{b}^H(\omega_l)\mathbf{Y}_l}{\mathbf{b}^H(\omega_l)\mathbf{b}(\omega_l)} = \frac{\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l)}{\| \mathbf{S} \|_F^2}, \tag{14}$$

where $(\cdot)^H$ represents the conjugate transpose and $\| \cdot \|_F$ denotes the *Frobenius* norm (Stewart, 1973). Then the estimate $\hat{\omega}_l$ of ω_l is obtained as follows:

$$\begin{aligned} \hat{\omega}_l &= \arg\min_{\omega_l} \left\| \mathbf{Y}_l - \frac{\mathbf{b}(\omega_l)\mathbf{b}^H(\omega_l)}{\mathbf{b}^H(\omega_l)\mathbf{b}(\omega_l)} \mathbf{Y}_l \right\|^2 = \arg\max_{\omega_l} \frac{\mathbf{Y}_l^H \mathbf{b}(\omega_l)\mathbf{b}^H(\omega_l)\mathbf{Y}_l}{\mathbf{b}^H(\omega_l)\mathbf{b}(\omega_l)} \\ &= \arg\max_{\omega_l} \left| \mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l) \right|^2, \end{aligned} \tag{15}$$

where we have used the fact that $\mathbf{b}^H(\omega_l)\mathbf{b}(\omega_l) = \| \mathbf{S} \|_F^2$ and hence is independent of ω_l. Hence $\hat{\omega}_l$ is obtained as the location of the dominant peak of the magnitude squared of the Fourier transform, $|\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l)|^2$, which can be efficiently computed by using the fast Fourier transform (FFT) with the weighted data vector $\mathbf{S}^*\mathbf{Y}_l$ padded with zeros. An alternative scheme to zero-padding FFT is to find an approximate peak location first by using FFT without much zero-padding and then perform a fine search nearby the approximate peak location by, for example, the *fmin* function in MATLAB, which uses the Golden section search algorithm. With the estimate of ω_l at hand, $\hat{\alpha}_l$ can be easily computed from the corresponding complex height:

$$\hat{\alpha}_l = \left. \frac{\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l)}{\| \mathbf{S} \|_F^2} \right|_{\omega_l = \hat{\omega}_l} . \tag{16}$$

With the above simple preparations, we now present the WRELAX algorithm.

Step (1): Assume $L = 1$. Obtain $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ from $\mathbf{Y}$ by using (15) and (16).

Step (2): Assume $L = 2$. Compute $\mathbf{Y}_2$ with (11) by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ obtained in Step (1). Obtain $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}$ from $\mathbf{Y}_2$. Next, compute $\mathbf{Y}_1$ by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}$ and then redetermine $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ from $\mathbf{Y}_1$.

Iterate the previous two substeps until "practical convergence" is achieved (to be discussed later on).

Step (3): Assume $L = 3$. Compute $\mathbf{Y}_3$ by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}^{2}$ obtained in Step (2). Obtain $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=3}$ from $\mathbf{Y}_3$. Next, compute $\mathbf{Y}_1$ by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}^{3}$ and redetermine $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ from $\mathbf{Y}_1$. Then compute $\mathbf{Y}_2$ by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1,3}$ and redetermine $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}$ from $\mathbf{Y}_2$.

Iterate the previous three substeps until "practical convergence".

Remaining Steps: Continue similarly until L is equal to the desired or estimated number of signals. (Whenever L is unknown, it can be estimated from the available data, for instance, by using the generalized Akaike information criterion (AIC) rules which are particularly tailored to the WRELAX method of parameter estimation. See, for example, (Li & Stoica, 1996).)

The "practical convergence" in the iterations of the above WRELAX method may be determined by checking the relative change of the cost function $C_1(\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}^{L})$ in (6) between two consecutive iterations. The algorithm is bound to converge to at least some local minimum point (Karmanov, 1977). The convergence speed depends on the time delay spacing of the signals. If the spacing between any two signals is larger than the reciprocal of the signal bandwidth, the algorithm converges in a few steps. As the spacing of the signals becomes closer, the convergence speed becomes slower.

Once we have obtained the estimates $\{\hat{\omega}_l\}_{l=1}^{L}$, the estimates $\{\hat{\tau}_l\}_{l=1}^{L}$ of $\{\tau_l\}_{l=1}^{L}$ can be determined by using (5).

At this point, we would like to point out the relationship between WRELAX and the conventional matched filter approach. The matched filter approach can also be formulated in the frequency domain. Let

$$F(\omega) = \left| \mathbf{a}^H(\omega)(\mathbf{S}^*\mathbf{Y}) \right|^2 . \tag{17}$$

The matched filter method searches for the L largest peak positions of $F(\omega)$ as the estimates of $\{\omega_l\}_{l=1}^{L}$, and then the gains are determined as follows

$$\hat{\alpha}_l = \left. \frac{\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y})}{\| \mathbf{S} \|_F^2} \right|_{\omega_l = \hat{\omega}_l} , \quad l = 1, 2, \cdots, L . \tag{18}$$

Hence when there is only one signal, this one-dimensional matched filter approach is equivalent to the WRELAX algorithm. However, when there are multiple signals that are not well separated, this conventional matched filter approach will perform poorly. In this case, a multidimensional matched filter method (Bell & Ewart, 1986) could be used and the method is equivalent to the NLS fitting approach (Bell & Ewart, 1986). The WRELAX algorithm decouples the multidimensional matched filters into a sequence of one-dimensional matched filters. Thus the excellent parameter estimation performance of the NLS fitting approach can be achieved at a much lower implementation cost.

Similar to the WRELAX algorithm, the EM algorithm proposed in (Feder & Weinstein, 1988) also transforms the multidimensional optimization problem into a series of one-dimensional optimization problems. The detailed implementations of the algorithms, however, are quite different. The EM algorithm consists of two steps, the E (Estimate) step and the M (Maximize) step. The idea is to decompose the observed data into their signal components (the E step) and then to estimate the parameters of each signal component separately (the M step). The

algorithm is iterative, using the current parameter estimates to decompose the observed data. At each E step, the residue error corresponding to the current estimates is also decomposed among different signal components. Although initial conditions are needed by EM, no systematic initialization method is given in (Feder & Weinstein, 1988). We have also found that the performance of EM is very sensitive to the initial conditions used. Even with the same initial conditions, our numerical examples show that the convergence speed of EM can be much slower than the last step of WRELAX. Further, WRELAX does not require any initial conditions before its iterations and the first $L-1$ steps of WRELAX can provide an excellent initial condition for Step L.

Consider next the case where $\{\alpha_l\}_{l=1}^{L}$ are real-valued. Minimizing $C_2(\alpha_l, \omega_l)$ with respect to α_l and ω_l yields

$$\hat{\alpha}_l = \left. \frac{\mathrm{Re}\left[\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l)\right]}{\| \mathbf{S} \|_F^2} \right|_{\omega_l=\hat{\omega}_l}, \tag{19}$$

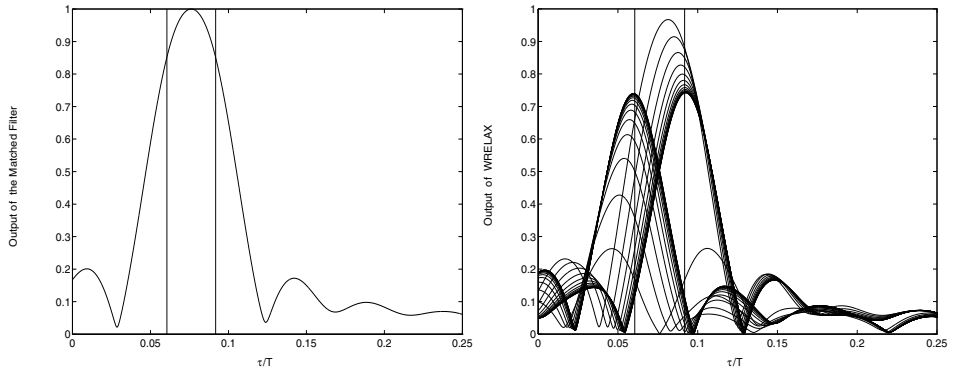
where $\mathrm{Re}(\mathbf{X})$ denotes the real part of $\mathbf{X}$, and

$$\hat{\omega}_l = \arg\max_{\omega_l} \mathrm{Re}^2 \left[\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l)\right]. \tag{20}$$

The WRELAX algorithm could also be implemented in the time domain, which is based on the correlations. However, we prefer to use the frequency domain version of WRELAX. For the time domain version, we could be restricted to use the discrete values of $\{\tau_l\}_{l=1}^{L}$ if we only know the sampled version of $s(t)$. For this case, if a more accurate delay estimate is required, then one has to resort to interpolation (Bell & Ewart, 1986). This inconvinence can be avoided by transforming the problem to the frequency domain, where $\{\tau_l\}_{l=1}^{L}$ can take on a continuum of values. Even without considering the additional interpolation cost, the computational load of the time domain correlation-based WRELAX is heavier than that of the frequency domain WRELAX.

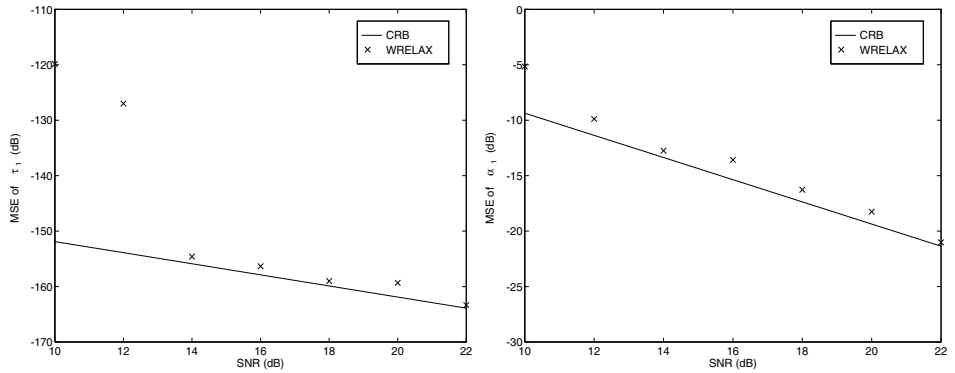
We present several numerical examples to demonstrate the performance of the proposed algorithms. In the following examples, we use a windowed chirp signal, $s(t) = w(t)e^{j\beta(t-\frac{T_0}{2})^2}$, $0 \le t \le T_0$, where β is the chirp rate and $w(t)$ is a bell-shaped window function. We use $N = 64$, $\beta = \pi \times 10^{12}$, the signal bandwidth $B_s = \beta T_0/\pi$, and the sampling frequency $f_s = 2B_s$. T_0 is chosen in such a way that $T_0 = (N/2-1)T_s$. The resolution limit of the conventional matched filter method is around $\tau_e = 1/B_s$. $L = 2$ signals are assumed to be superimposed together with $\alpha_1 = e^{j\pi/8}$, $\alpha_2 = e^{j\pi/4}$.

In the first example, the additive noise is zero-mean white Gaussian and SNR=SNR1=SNR2=10dB. The time delay spacing between the two signals is $\tau_2 - \tau_1 = \tau_e$. Even in this case, the conventional matched filter method fails to resolve the two signals, as can be seen from Figure 1(a), where the horizontal axis denotes the normalized time delay τ/T and the two vertical lines indicate the true time delays of the two signals. However, using WRELAX we can resolve them very well. As pointed out before, the WRELAX algorithm can be viewed as transforming the multi-dimensional matched filters into a sequence of one-dimensional matched filters. The outputs of the two matched filters for all iterations are plotted in Figure 1(b), which illustrates the convergence process of WRELAX. At the beginning of the iteration, the peak positions and the corresponding gain estimates obtained from the filter outputs differ from their true values (one gain estimate is larger and the other



(a) (b)

Fig. 1. Illustrative comparison of WRELAX with the matched filter method. The vertical lines denote the true time delay. (a) The output of the matched filter and (b)The outputs of the two decoupled matched filters with WRELAX for all iterations.



(a) (b)

Fig. 2. MSEs ("×") of WRELAX and Cramér-Rao bound (CRB) (solid line) for (a) τ_1 and (b) α_1.

is smaller than the corresponding true gains). After several steps, they converge to the true time delays and gains.

In this example, the time delay spacing between the two signals is $\tau_2 - \tau_1 = 0.5\tau_e$. The MSEs for the first signal using WRELAX are compared with the corresponding CRBs in Figure 2 and the MSE and CRB curves for the other signal are similar. From Figure 2, it can be noted that the MSEs obtained by using WRELAX approach the corresponding CRBs as the SNR increases.

3.2 *T*oeplitz property based *W*eighted Fourier transform and *RELAX*ation (TWRELAX) algorithm

Here, we extend the above WRELAX algorithm to the case of multiple looks. Two scenarios will be considered, which include 1) fixed delays but arbitrary gains and 2) fixed delays and

fixed gains. In radar applications, the two cases correspond to two different target fluctuation models, e.g., Swerling I and Swerling II (Barton, 1988).

3.2.1 Fixed delays but arbitrary gains

Consider the case where multiple pulses are transmitted and the ranges of target scatterers remain the same but their gains change randomly during the observation interval.

Let $\mathbf{Y}^{(m)}$ be the DFT of the received vector due to the mth pulse. Then

$$\mathbf{Y}^{(m)} = \sum_{l=1}^{L} \alpha_l^{(m)} [\mathbf{S}\mathbf{a}(\omega_l)] + \mathbf{E}^{(m)}, \quad m = 1, 2, ..., M, \tag{21}$$

where $\alpha_l^{(m)}$ denotes the gain of the lth scatterers due to the mth pulse and the noise vectors $\{\mathbf{E}^{(m)}\}_{m=1}^{M}$ are assumed independent of each other. Our problem of interest is to estimate $\left\{\alpha_l^{(m)}, \omega_l\right\}_{l=1,..,L;m=1,...,M}$ from $\left\{\mathbf{Y}^{(m)}\right\}_{m=1}^{M}$.

We now extend the WRELAX algorithm to this multiple look case. The extended WRELAX algorithm minimizes the following NLS criterion:

$$C_3(\left\{\alpha_l^{(m)}, \omega_l\right\}_{l=1,..,L;m=1,...,M}) = \sum_{m=1}^{M} \left\| \mathbf{Y}^{(m)} - \sum_{l=1}^{L} \alpha_l^{(m)} \mathbf{b}(\omega_l) \right\|^2, \tag{22}$$

where $\mathbf{b}(\omega_l)$ is defined in (12).

Before we present the extended WRELAX algorithm, let us consider the following preparations. Let

$$\mathbf{Y}_l^{(m)} = \mathbf{Y}^{(m)} - \sum_{i=1, i\neq l}^{L} \hat{\alpha}_i^{(m)} \mathbf{b}(\hat{\omega}_i), \quad m = 1, ..., M, \tag{23}$$

where $\left\{\hat{\alpha}_i^{(m)}, \hat{\omega}_i\right\}_{i=1,...,L,i\neq l,m=1,...,M}$ are assumed given. Then the cost function $C_3(\left\{\alpha_l^{(m)}, \omega_l\right\}_{l=1,..,L;m=1,...,M})$ becomes

$$C_4(\{\alpha_l^{(m)}\}_{m=1}^{M}, \omega_l) = \sum_{m=1}^{M} \left\| \mathbf{Y}_l^{(m)} - \alpha_l^{(m)} \mathbf{b}(\omega_l) \right\|^2. \tag{24}$$

Minimizing $C_4(\{\alpha_l\}_{l=1}^{L}, \omega_l)$ with respect to the complex-valued $\{\alpha_l^{(m)}\}_{m=1}^{M}$ and ω_l yields

$$\hat{\alpha}_l^{(m)} = \left. \frac{\mathbf{a}^H(\omega_l)(\mathbf{S}^* \mathbf{Y}_l^{(m)})}{\| \mathbf{S} \|_F^2} \right|_{\omega_l = \hat{\omega}_l}, \tag{25}$$

and

$$\hat{\omega}_l = \arg\max_{\omega_l} \left[\sum_{m=1}^{M} \left| \mathbf{a}^H(\omega_l)(\mathbf{S}^* \mathbf{Y}_l^{(m)}) \right|^2 \right]. \tag{26}$$

Minimizing $C_4(\{\alpha_l^{(m)}\}_{m=1}^M, \omega_l)$ with respect to the real-valued $\left\{\alpha_l^{(m)}\right\}_{m=1}^M$ and ω_l yields

$$\hat{\alpha}_l^{(m)} = \left. \frac{\mathrm{Re}\left[\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l^{(m)})\right]}{\| \mathbf{S} \|_F^2} \right|_{\omega_l=\hat{\omega}_l}, \tag{27}$$

and

$$\hat{\omega}_l = \arg\max_{\omega_l} \left\{ \sum_{m=1}^{M} \mathrm{Re}^2 \left[\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l^{(m)})\right] \right\}. \tag{28}$$

3.2.2 Fixed delays and fixed gains

When both the delays and gains of the target scatterers remain the same during the multiple look interval, we can derive an ML estimator when the noise is assumed to be a zero-mean colored Gaussian noise with an unknown covariance matrix $\mathbf{Q}$. Note that although we could continue to use the NLS approach for the current problem, we prefer to take the noise statistics into account since we will show below that doing so in this case introduces little difficulties for sufficiently large M. For the former problems, modeling the noise with an unknown covariance matrix $\mathbf{Q}$ makes the ML approach ill-defined due to too many unknowns (Li et al. b, 2002).

Let $\mathbf{Y}^{(m)}$ be the DFT of the received data vector due to the mth pulse which can be written as

$$\mathbf{Y}^{(m)} = \sum_{l=1}^{L} \alpha_l \mathbf{b}(\omega_l) + \mathbf{E}^{(m)}, \quad m = 1, ..., M, \tag{29}$$

where the noise vectors $\{\mathbf{E}^{(m)}\}_{m=1}^M$ are assumed to be zero-mean colored Gaussian random vectors with an unknown covariance matrix $\mathbf{Q}$ that are independent of each other. Let

$$\mathbf{b} = [\, \mathbf{b}(\omega_1)\ \mathbf{b}(\omega_2)\ \cdots\ \mathbf{b}(\omega_L)\,]^T, \tag{30}$$

and

$$\boldsymbol{\alpha} = [\, \alpha_1\ \alpha_2\ \cdots\ \alpha_L\,]^T. \tag{31}$$

Then

$$\mathbf{Y}^{(m)} = \mathbf{b}\boldsymbol{\alpha} + \mathbf{E}^{(m)}, \quad m = 1, 2, ..., M. \tag{32}$$

The log-likelihood function of $\mathbf{Y}^{(m)}$ is proportional to (within an additive constant):

$$-\ln[\det(\mathbf{Q})] - \mathrm{tr}\left\{ \mathbf{Q}^{-1} \frac{1}{M} \sum_{m=1}^{M} \left[\mathbf{Y}^{(m)} - \mathbf{b}\boldsymbol{\alpha}\right] \left[\mathbf{Y}^{(m)} - \mathbf{b}\boldsymbol{\alpha}\right]^H \right\}, \tag{33}$$

where $\det(\cdot)$ denotes the determinant of a matrix and $\mathrm{tr}(\cdot)$ denotes the trace of a matrix. Consider first the estimate of $\mathbf{Q}$ and the unstructured estimate of $\mathbf{C} = \mathbf{b}\boldsymbol{\alpha}$. It is easy to show

that the estimate $\hat{\mathbf{Q}}$ of $\mathbf{Q}$ is

$$\hat{\mathbf{Q}} = \frac{1}{M}\sum_{m=1}^{M}\left[\mathbf{Y}^{(m)} - \mathbf{C}\right]\left[\mathbf{Y}^{(m)} - \mathbf{C}\right]^H, \tag{34}$$

where $\hat{\mathbf{C}}$ may be obtained by minimizing the following cost function:

$$C_5 = \det\left[\frac{1}{M}\sum_{m=1}^{M}\left(\mathbf{Y}^{(m)} - \mathbf{C}\right)\left(\mathbf{Y}^{(m)} - \mathbf{C}\right)^H\right]. \tag{35}$$

Let

$$\hat{\mathbf{R}}_{Y1} = \frac{1}{M}\sum_{m=1}^{M}\mathbf{Y}^{(m)}, \tag{36}$$

and

$$\hat{\mathbf{R}}_{YY} = \frac{1}{M}\sum_{m=1}^{M}\mathbf{Y}^{(m)}(\mathbf{Y}^{(m)})^H. \tag{37}$$

Then

$$\begin{aligned}\mathbf{G} &= \frac{1}{M}\sum_{m=1}^{M}\left[\mathbf{Y}^{(m)} - \mathbf{C}\right]\left[\mathbf{Y}^{(m)} - \mathbf{C}\right]^H \\ &= \hat{\mathbf{R}}_{YY} - \mathbf{C}\hat{\mathbf{R}}_{Y1}^H - \hat{\mathbf{R}}_{Y1}\mathbf{C}^H + \mathbf{C}\mathbf{C}^H \\ &= \left[\mathbf{C} - \hat{\mathbf{R}}_{Y1}\right]\left[\mathbf{C} - \hat{\mathbf{R}}_{Y1}\right]^H + \hat{\mathbf{R}}_{YY} - \hat{\mathbf{R}}_{Y1}\hat{\mathbf{R}}_{Y1}^H.\end{aligned} \tag{38}$$

To minimize $\det(\mathbf{G})$, we have

$$\hat{\mathbf{C}} = \hat{\mathbf{R}}_{Y1}. \tag{39}$$

Then using the $\hat{\mathbf{C}}$ in (39) to replace the $\mathbf{C}$ in (34) yields

$$\hat{\mathbf{Q}} = \hat{\mathbf{R}}_{YY} - \hat{\mathbf{R}}_{Y1}\hat{\mathbf{R}}_{Y1}^H. \tag{40}$$

With these notations, the above C_5 can be rewritten as

$$\begin{aligned}C_5 &= \det\left[\hat{\mathbf{R}}_{YY} - \mathbf{C}\hat{\mathbf{R}}_{Y1}^H - \hat{\mathbf{R}}_{Y1}\mathbf{C}^H + \mathbf{C}\mathbf{C}^H\right] = \det\left[\hat{\mathbf{R}}_{YY} - \hat{\mathbf{R}}_{Y1}\hat{\mathbf{R}}_{Y1}^H + (\mathbf{C} - \hat{\mathbf{C}})(\mathbf{C} - \hat{\mathbf{C}})^H\right] \\ &= \det(\hat{\mathbf{Q}})\det\left[\mathbf{I} + \hat{\mathbf{Q}}^{-1}(\mathbf{C} - \hat{\mathbf{C}})(\mathbf{C} - \hat{\mathbf{C}})^H\right] = \det(\hat{\mathbf{Q}})\left[1 + (\mathbf{C} - \hat{\mathbf{C}})^H\hat{\mathbf{Q}}^{-1}(\mathbf{C} - \hat{\mathbf{C}})\right],\end{aligned} \tag{41}$$

where we have used the fact that $\det(\mathbf{I} + \mathbf{ab}) = \det(\mathbf{I} + \mathbf{ba})$ if the dimensions of $\mathbf{a}$ and $\mathbf{b}$ permit. Hence minimizing C_5 is equivalent to minimizing

$$C_6(\{\alpha_l, \omega_l\}_{l=1}^{L}) = \left[\mathbf{C} - \hat{\mathbf{C}}\right]^H\hat{\mathbf{Q}}^{-1}\left[\mathbf{C} - \hat{\mathbf{C}}\right] = \left[\mathbf{b}\boldsymbol{\alpha} - \hat{\mathbf{C}}\right]^H\hat{\mathbf{Q}}^{-1}\left[\mathbf{b}\boldsymbol{\alpha} - \hat{\mathbf{C}}\right], \tag{42}$$

which is again a highly nonlinear optimization problem.

We consider below using the relaxation based approach to minimize $C_6(\{\alpha_l, \omega_l\}_{l=1}^L)$. Let

$$\hat{\mathbf{C}}_l = \hat{\mathbf{C}} - \sum_{i=1, i\neq l}^{L} \hat{\alpha}_i \mathbf{b}(\hat{\omega}_i), \tag{43}$$

where $\{\hat{\alpha}_i, \hat{\omega}_i\}_{i=1, i\neq l}^L$ are assumed given. Then minimizing C_6 becomes minimizing

$$C_7(\alpha_l, \omega_l) = \left[\hat{\mathbf{C}}_l - \alpha_l \mathbf{b}(\omega_l)\right]^H \hat{\mathbf{Q}}^{-1} \left[\hat{\mathbf{C}}_l - \alpha_l \mathbf{b}(\omega_l)\right]. \tag{44}$$

Consider first the case of complex-valued $\{\alpha_l\}_{l=1}^L$. Minimizing $C_7(\alpha_l, \omega_l)$ with respect to α_l and ω_l yields:

$$\hat{\alpha}_l = \left.\frac{\mathbf{b}^H \hat{\mathbf{Q}}^{-1} \hat{\mathbf{C}}_l}{\mathbf{b}^H(\omega_l) \hat{\mathbf{Q}}^{-1} \mathbf{b}(\omega_l)}\right|_{\omega_l = \hat{\omega}_l} = \left.\frac{\mathbf{a}^H(\omega_l) \left(\mathbf{S}^* \hat{\mathbf{Q}}^{-1} \hat{\mathbf{C}}_l\right)}{\left\|\hat{\mathbf{Q}}^{-\frac{1}{2}} \mathbf{S}\mathbf{a}(\omega_l)\right\|^2}\right|_{\omega_l = \hat{\omega}_l}, \tag{45}$$

and

$$\hat{\omega}_l = \arg\max_{\omega_l} \frac{\left|\mathbf{a}^H(\omega_l) \mathbf{S}^* \hat{\mathbf{Q}}^{-1} \hat{\mathbf{C}}_l\right|^2}{\left\|\hat{\mathbf{Q}}^{-\frac{1}{2}} \mathbf{S}\mathbf{a}(\omega_l)\right\|^2}. \tag{46}$$

Consider next the case of real-valued $\{\alpha_l\}_{l=1}^L$. Minimizing $C_7(\alpha_l, \omega_l)$ with respect to α_l and ω_l yields

$$\hat{\alpha}_l = \left.\frac{\mathrm{Re}\left[\mathbf{a}^H(\omega_l) \mathbf{S}^* \hat{\mathbf{Q}}^{-1} \hat{\mathbf{C}}_l\right]}{\left\|\hat{\mathbf{Q}}^{-\frac{1}{2}} \mathbf{S}\mathbf{a}(\omega_l)\right\|^2}\right|_{\omega_l = \hat{\omega}_l}, \tag{47}$$

and

$$\hat{\omega}_l = \arg\max_{\omega_l} \frac{\mathrm{Re}^2\left[\mathbf{a}^H(\omega_l) \mathbf{S}^* \hat{\mathbf{Q}}^{-1} \hat{\mathbf{C}}_l\right]}{\left\|\hat{\mathbf{Q}}^{-\frac{1}{2}} \mathbf{S}\mathbf{a}(\omega_l)\right\|^2}. \tag{48}$$

In the above derivations, $\hat{\mathbf{Q}}^{-1}$ plays the role of whitening the noise. A good estimate of $\mathbf{Q}$ requires a large number of independent data vectors (i.e., M should be large enough as compared with N). When M is small, the noise covariance matrix estimated from (40) is singular or near singular. At least N data vectors are needed to guarantee that the matrix $\hat{\mathbf{Q}}$ is non-singular with probability one.

Usually, the receiver noise $e(t)$ in (1) can be modeled as a zero-mean stationary and ergodic Gaussian stochastic process. Let the covariance matrix corresponding to the sampled noise vector $[e(0),\ e(T_s)\ \cdots\ e((N-1)T_s)]^T$ be $\mathbf{Q}_t$, where the subscript "t" represents the covariance matrix of the noise in the time domain. Then $\mathbf{Q}_t$ is a Hermitian and Toeplitz matrix. The frequency domain noise covariance matrix $\mathbf{Q}$ is related to $\mathbf{Q}_t$ as follows

$$\mathbf{Q} = \boldsymbol{\gamma}\, \mathbf{Q}_t\, \boldsymbol{\gamma}^H, \tag{49}$$

where $\boldsymbol{\gamma}$ is the DFT matrix,

$$\boldsymbol{\gamma} = \frac{1}{\sqrt{N}} \left[\mathbf{a}(-\pi) \ \mathbf{a}(-\pi + 2\pi/N) \ \cdots \ \mathbf{a}(\pi - 2\pi/N) \right]^H, \tag{50}$$

and

$$\mathbf{a}(\omega) = \left[e^{j\omega(-N/2)} \ e^{j\omega(-N/2+1)} \ \ldots \ e^{j\omega(N/2-1)} \right]^T . \tag{51}$$

It can be shown that, in general, $\mathbf{Q}$ is no longer a Toeplitz matrix. However, we can use the Toeplitz property of $\mathbf{Q}_t$ to improve the estimation performance. First, we can obtain $\hat{\mathbf{Q}}$ by using (40). Then the estimate $\hat{\mathbf{Q}}_t$ of $\mathbf{Q}_t$ can be obtained by using (49), which is

$$\hat{\mathbf{Q}}_t = \boldsymbol{\gamma}^H \hat{\mathbf{Q}} \boldsymbol{\gamma}. \tag{52}$$

Due to a finite number of data vectors, $\hat{\mathbf{Q}}_t$ is no longer a Toeplitz matrix. Although there are many ways to to modify $\hat{\mathbf{Q}}_t$ to obtain a Toeplitz matrix $\hat{\mathbf{Q}}_t^{(T)}$, in this paper we use the following simple approach. Let $\hat{q}_t(i,j)$ be the (i,j)th element of $\hat{\mathbf{Q}}_t$. Define

$$\hat{r}(k) = \frac{1}{N-k} \sum_{i=1}^{N-k} \hat{q}_t(i, i+k), \quad k = 0, 1, \cdots, N-1. \tag{53}$$

Then

$$\hat{\mathbf{Q}}_t^{(T)} = \begin{bmatrix} \hat{r}(0) & \hat{r}(1) & \cdots & \hat{r}(N-1) \\ \hat{r}^*(1) & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \hat{r}(1) \\ \hat{r}^*(N-1) & \cdots & \hat{r}^*(1) & \hat{r}(0) \end{bmatrix}. \tag{54}$$

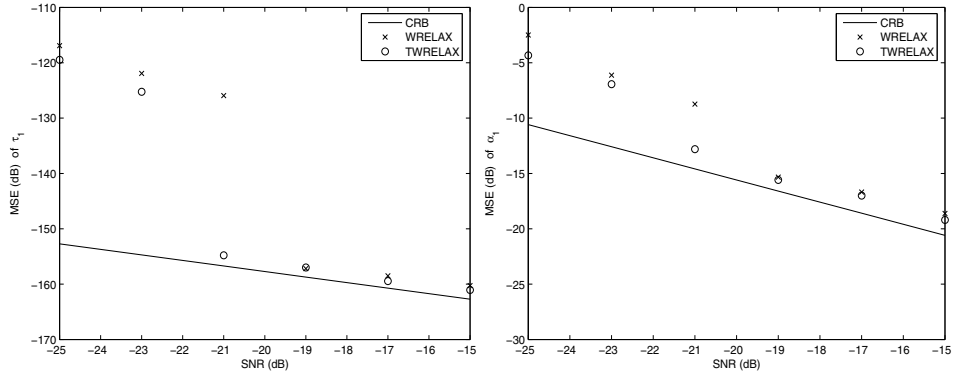
Using $\hat{\mathbf{Q}}^{(T)}$ instead of $\hat{\mathbf{Q}}$ in (45)-(48), where

$$\hat{\mathbf{Q}}^{(T)} = \boldsymbol{\gamma} \hat{\mathbf{Q}}_t^{(T)} \boldsymbol{\gamma}^H, \tag{55}$$

we obtain a new algorithm referred to as TWRELAX. The TWRELAX algorithm can greatly improve the estimation performance of WRELAX, especially when M is small as compared with N, as can be seen from the numerical examples below.

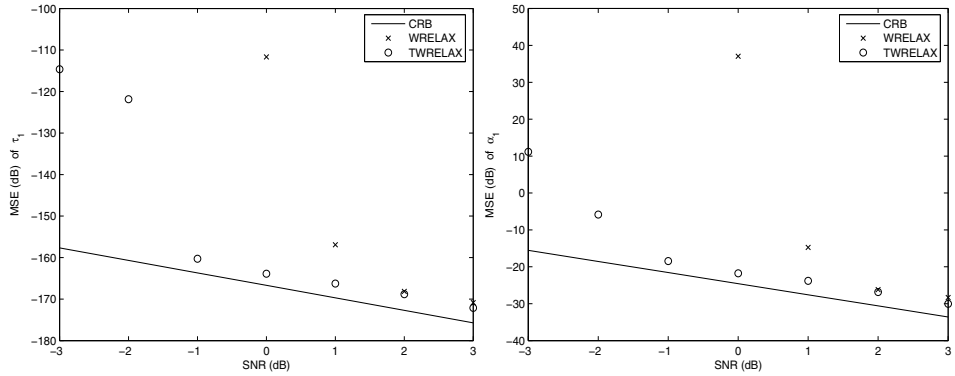
For notational convenience, we denote the new algorithms with and without exploiting the Toeplitz property of the noise covariance matrix by TWRELAX and WRELAX, respectively. Here, the time delay spacing between the two signals is $\tau_2 - \tau_1 = 0.5\tau_e$.We have used $\epsilon = 0.001$ to test the convergence of TWRELAX and WRELAX. The colored noise is modeled as a first-order autoregressive (AR) process with coefficient a_1=-0.85.

The MSEs of TWRELAX ("$\circ$") and WRELAX ("$\times$") are compared with the corresponding CRBs (solid line) in Figures 3 and 4 as a function of the SNR and the normalized look number $\log_2(M/N)$, respectively. We fix $M = 4N$ in Figure 3 and SNR=-5 dB in Figure 4. From Figure 3, it can be noted that, when M is sufficiently large, both TWRELAX and WRELAX can approach the CRBs for a wide range of SNRs and the former performs slightly better than the latter. However, when M is small as compared to N, TWRELAX outperforms WRELAX significantly, as can be seen from Figure 4. For this example, when $M = N/2$, the MSEs of TWRELAX are very close to the CRBs, while $M = 4N$ is required before the MSEs of WRELAX



(a) (b)

Fig. 3. MSEs of WRELAX ("×"), TWRELAX ("○"), and CRBs (solid line) for (a) τ_1 , (b) α_1 as a function of SNR with $M = 4N$.



(a) (b)

Fig. 4. MSEs of WRELAX ("×"), TWRELAX ("○"), and CRBs (solid line) for (a) τ_1 , (b) α_1 as a function of the normalized look number $\log_2(M/N)$ with SNR=−5 dB.

approach the CRBs. Note also that in Figure 4, due to the inversion of poorly estimated noise covariance matrices, some points of the MSE curves of WRELAX are beyond the scope of the axis limits.

3.3 Hybrid-WRELAX algorithm and *EX*tended *I*nvariance *P*rinciple Based *WRELAX* (EXIP-WRELAX) algorithm

When bandpass real-valued probe signals are used, the correlation function between the received and the known transmitted signals oscillates near the carrier frequency of the transmitted signal. In this case, many existing time delay estimation algorithms perform poorly due to converging to local optimum points. Here, two efficient algorithms are proposed to deal with the above problem. First we assume that the signal amplitudes are complex-valued and use WRELAX to obtain the initial estimates of the delays and the

amplitudes of the superimposed signals by minimizing a much smoother NLS cost function. Then the initial estimates are refined with two approaches. One approach (referred to as Hybrid-WRELAX) uses the last step of the WRELAX algorithm to minimize the true NLS cost function corresponding to the real-valued signal amplitudes. The other approach (referred to as EXIP-WRELAX) uses the extended invariance principle (EXIP). For Hybrid-WRELAX, the refinement step is iterative, while it is not for EXIP-WRELAX.

Real-valued signals are often bandpass signals that occur, for example, in underwater sonar and ultra wideband ground penetrating radar applications. Bandpass signals have highly oscillatory correlation functions, which makes the super resolution time delay estimation problem more difficult. The larger the center frequency of the pass band, the sharper the oscillation of the correlation function.

The same data model as that in the Section 2 is adopted. However, the transmitted signal $s(t)$ and the received signal $y(t)$ are real-valued, and the gains $\{\alpha_l\}_{l=1}^{L}$ and the noise $e(t)$ are also real-valued.

3.3.1 Hybrid-WRELAX

Since both the transmitted signal $s(t)$ and the received signal $y(t)$ are real-valued, their Fourier transforms are conjugate symmetric, i.e., $Y(-k) = Y^*(k)$ and $S(-k) = S^*(k), \quad k = 1, 2, \cdots, N/2-1$, where $(\cdot)^*$ denotes the complex conjugate, and $Y(-N/2)$, $Y(0)$, $S(-N/2)$, and $S(0)$ are real-valued. It can be readily shown that the cost function (6) is equivalent to

$$C_8(\{\alpha_l, \omega_l\}_{l=1}^{L}) = \sum_{k=-N/2}^{0} W^2(k) \left| Y(k) - S(k) \sum_{l=1}^{L} \alpha_l e^{j\omega_l k} \right|^2, \tag{56}$$

where $\{W(k) = 1\}_{k=-N/2+1}^{-1}$ and $W(-N/2) = W(0) = 1/\sqrt{2}$. We assume that $e(nT_s)$ is a real-valued zero-mean white Gaussian random process with variance σ^2. Yet $E(k)$ will not be a circularly symmetric complex-valued zero-mean white Gaussian random process since $E(-k) = E^*(k), \quad k = 1, 2, \cdots, N/2-1$. (The circularly symmetric assumption on the noise is widely used in the literature (Stoica & Moses, 1997).)

The cost function $C_8(\{\alpha_l, \omega_l\}_{l=1}^{L})$ in (56) with $\{\alpha_l\}_{l=1}^{L}$ being real-valued is referred to as the true cost function. Minimizing $C_8(\{\alpha_l, \omega_l\}_{l=1}^{L})$ with respect to the unknown parameters is a highly nonlinear optimization problem. For narrowband transmitted signals, the cost function is highly oscillatory and have numerous closely spaced local minima, which makes it very difficult to find the global minimum. By assuming the real-valued amplitudes $\{\alpha_l\}_{l=1}^{L}$ to be complex-valued, a much smoother cost function can be obtained. This is equivalent to formulate the original time delay estimation problem in its complex analytic signal form. Since the analytic signal of the transmitted signal is lowpass, its autocorrelation function is no longer oscillatory. This is the conventional complex demodulation process and is widely used in practice. Although it is much easier to find the global minimum of the cost function corresponding to complex-valued amplitudes, the so-obtained estimates can be much less accurate than those obtained by minimizing the true cost function. The two cost functions share the same global minimum only when there is no noise. However, as suggested in (Manickam et al., 1994; Vaccaro et al., 1992), we can minimize the cost function associated with complex-valued amplitudes to obtain the initial conditions needed to minimize the true cost function. Below, we present a relaxation based global minimizer of the NLS criterion

based on this idea. The algorithm is referred to as the Hybrid-WRELAX algorithm. It simply requires a sequence of weighted Fourier transforms.

Before we present our approach, let us consider the following preparations. Let

$$\begin{aligned}\bar{\mathbf{W}} &= \text{diag}\left\{ W(-N/2), W(-N/2+1), \cdots, W(-1), W(0) \right\} \\ &= \text{diag}\left\{ \tfrac{1}{\sqrt{2}}, 1, \cdots, 1, \tfrac{1}{\sqrt{2}} \right\}, \end{aligned} \tag{57}$$

$$\bar{\mathbf{Y}} = \bar{\mathbf{W}} \left[Y(-N/2)\ Y(-N/2+1) \cdots Y(0) \right]^T, \tag{58}$$

$$\bar{\mathbf{S}} = \bar{\mathbf{W}}\ \text{diag}\left\{ S(-N/2), S(-N/2+1), \cdots, S(0) \right\}, \tag{59}$$

and

$$\bar{\mathbf{a}}(\omega_l) = \left[e^{j\omega_l(-N/2)}\ e^{j\omega_l(-N/2+1)} \cdots 1 \right]^T. \tag{60}$$

Denote

$$\bar{\mathbf{Y}}_l = \bar{\mathbf{Y}} - \sum_{i=1, i\neq l}^{L} \hat{\alpha}_i [\bar{\mathbf{S}}\bar{\mathbf{a}}(\hat{\omega}_i)] \tag{61}$$

where $\{\hat{\alpha}_i, \hat{\omega}_i\}_{i=1,i\neq l}^{L}$ are assumed to be given. Let

$$\bar{\mathbf{b}}(\omega_l) = \bar{\mathbf{S}}\bar{\mathbf{a}}(\omega_l), \quad l = 1, 2, \cdots, L. \tag{62}$$

Then (56) becomes

$$C_9(\alpha_l, \omega_l) = \| \bar{\mathbf{Y}}_l - \alpha_l \mathbf{b}_h(\omega_l) \|^2, \tag{63}$$

where $\| \cdot \|$ denotes the Euclidean norm. Minimizing $C_9(\alpha_l, \omega_l)$ with respect to the real-valued α_l yields the estimate $\hat{\alpha}_l$ of α_l

$$\begin{aligned}\hat{\alpha}_l &= \frac{\text{Re}\left[\bar{\mathbf{b}}^H(\omega_l)\bar{\mathbf{Y}}_l\right]}{\bar{\mathbf{b}}^H(\omega_l)\bar{\mathbf{b}}(\omega_l)} \\ &= \frac{\text{Re}\left[\bar{\mathbf{a}}^H(\omega_l)(\bar{\mathbf{S}}^*\bar{\mathbf{Y}}_l)\right]}{\| \bar{\mathbf{S}} \|_F^2}, \end{aligned} \tag{64}$$

where $(\cdot)^H$ denotes the conjugate transpose, $\text{Re}(\mathbf{Z})$ represents the real part of $\mathbf{Z}$, and $\| \cdot \|_F$ denotes the *Frobenius* norm (Stewart, 1973). (More specifically, $\| \mathbf{S} \|_F = \sqrt{\sum_{n=-N/2}^{0} |W(n)S(n)|^2}$.) Then the estimate $\hat{\omega}_l$ of ω_l is obtained as follows:

$$\begin{aligned}\hat{\omega}_l &= \arg\min_{\omega_l} \left\| \bar{\mathbf{Y}}_l - \frac{\text{Re}\left[\bar{\mathbf{b}}^H(\omega_l)\bar{\mathbf{Y}}_l\right]}{\bar{\mathbf{b}}^H(\omega_l)\bar{\mathbf{b}}(\omega_l)} \bar{\mathbf{b}}(\omega_l) \right\|^2 \\ &= \arg\max_{\omega_l} \text{Re}^2 \left[\bar{\mathbf{a}}^H(\omega_l)(\bar{\mathbf{S}}^*\bar{\mathbf{Y}}_l) \right], \end{aligned} \tag{65}$$

where we have used the fact that $\bar{\mathbf{b}}^H(\omega_l)\bar{\mathbf{b}}(\omega_l) = \sum_{n=-N/2}^{0}|W(n)S(n)|^2$ and hence is independent of ω_l. Hence $\hat{\omega}_l$ is obtained as the location of the dominant peak of $\mathrm{Re}^2\left[\bar{\mathbf{a}}^H(\omega_l)(\bar{\mathbf{S}}^*\bar{\mathbf{Y}}_l)\right]$. With the estimate of ω_l at hand, $\hat{\alpha}_l$ is easily computed from the corresponding complex height by using $\hat{\omega}_l$ to replace ω_l in (64).

Similarly, minimizing $C_9(\alpha_l, \omega_l)$ with respect to ω_l and the complex-valued α_l, respectively, yields the estimates $\hat{\omega}_l$ of ω_l and $\hat{\alpha}_l$ of α_l,

$$\hat{\omega}_l = \arg\max_{\omega_l} \left|\mathbf{a}^H(\omega_l)(\bar{\mathbf{S}}^*\bar{\mathbf{Y}}_l)\right|^2, \tag{66}$$

and

$$\hat{\alpha}_l = \left.\frac{\mathbf{a}^H(\omega_l)(\bar{\mathbf{S}}^*\bar{\mathbf{Y}}_l)}{\|\bar{\mathbf{S}}\|_F^2}\right|_{\omega_l=\hat{\omega}_l}, \tag{67}$$

where $\hat{\omega}_l$ can also be found via FFT with the weighted data vector.

With the above simple preparations, we now present the Hybrid-WRELAX algorithm.

Step 1: Obtain the initial conditions for Step 2 by assuming that $\{\alpha_l\}_{l=1}^{L}$ are complex-valued and using the WRELAX algorithm as follows:

Substep (1): Assume $L = 1$. Obtain $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ from $\mathbf{Y}$ by using (66) and (67).

Substep (2): Assume $L = 2$. Compute $\mathbf{Y}_2$ with (96) by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ obtained in Substep (1). Obtain $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}$ from $\mathbf{Y}_2$. Next, compute $\mathbf{Y}_1$ by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}$ and then redetermine $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ from $\mathbf{Y}_1$.

Iterate the update of $\{\hat{\omega}_2, \hat{\alpha}_2\}$ and $\{\hat{\omega}_1, \hat{\alpha}_1\}$ until "practical convergence" is achieved (to be discussed later on).

Substep (3): Assume $L = 3$. Compute $\mathbf{Y}_h$ by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}^{2}$ obtained in Substep (2). Obtain $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=3}$ from $\mathbf{Y}_3$. Next, compute $\mathbf{Y}_1$ by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}^{3}$ and redetermine $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ from $\mathbf{Y}_1$. Then compute $\mathbf{Y}_2$ by using $\{\hat{\omega}_1, \hat{\alpha}_1\}_{l=1,3}$ and redetermine $\{\hat{\omega}_1, \hat{\alpha}_1\}_{l=2}$ from $\mathbf{Y}_2$.

Iterate the update of $\{\hat{\omega}_3, \hat{\alpha}_3\}$,$\{\hat{\omega}_1, \hat{\alpha}_1\}$, and $\{\hat{\omega}_2, \hat{\alpha}_2\}$ until "practical convergence".

Remaining Substeps: Continue similarly until L is equal to the desired or estimated number of signals.

Step 2: Refine the estimates obtained in Step 1 with the last step of the WRELAX algorithm (i.e., the last substep of Step 1 above) by using Equations (64) and (65) derived for the real-valued $\{\alpha_l\}_{l=1}^{L}$ and using $\{\hat{\omega}_l\}_{l=1}^{L}$ and the real parts of $\{\hat{\alpha}_l\}_{l=1}^{L}$ obtained in Step 1 as initial conditions. Iteratively update $\{\hat{\omega}_l, \hat{\alpha}_l\}$, $l = 1, 2, \cdots, L$, until "practical convergence".

Note that WRELAX can be used directly for signals with real-valued amplitudes. For this case, the approach would consist of the substeps of Step 1 above except that (64) and (65) will be used instead of (66) and (67), respectively. We will use a numerical example in Section 5 to show the problem encountered by the direct use of WRELAX when the cost function is highly oscillatory.

3.3.2 EXIP-WRELAX

The Invariance Principle (IP) of ML estimators is well known in the estimation theory (Zehna, 1966). The invariance principle gives a simple answer to the relationship between the

minimizers of a given cost function parameterized in two different ways in some special cases. By appropriately reparameterizing the original cost function and enlarging the supporting domain of the parameter space, less accurate estimates can be obtained from this simple data model. These estimates may be refined to asymptotically achieve the performance available using the original data model. This is the basic idea behind the Extended Invariance Principle (EXIP) proposed in (Söderström & Stoica a, 1989) and (Söderström & Stoica b, 1989) for the purpose of achieving some computational advantages. In this section, we present an EXIP based algorithm, referred to as the EXIP-WRELAX algorithm, that avoids dealing with the highly oscillatory true cost function entirely.

By using (58) and (62), the cost function (56) with $\{\alpha_l\}_{l=1}^L$ being real-valued can be written in the following vector form

$$C_{10}(\boldsymbol{\eta}) = \left\| \bar{\mathbf{Y}} - \sum_{l=1}^{L} \alpha_l \bar{\mathbf{b}}(\omega_l) \right\|_2^2, \tag{68}$$

where

$$\boldsymbol{\eta} = \left[\boldsymbol{\alpha}^T \ \boldsymbol{\omega}^T \right]^T, \tag{69}$$

with

$$\boldsymbol{\alpha} = \left[\alpha_1 \ \alpha_2 \ \cdots \ \alpha_L \right]^T, \tag{70}$$

$$\boldsymbol{\omega} = \left[\omega_1 \ \omega_2 \ \cdots \ \omega_L \right]^T. \tag{71}$$

By replacing the real-valued amplitudes $\{\alpha_l\}_{l=1}^L$ with the complex-valued amplitudes $\{\tilde{\alpha}_l\}_{l=1}^L$ (notations introduced for the sake of clarity) in (68), we obtain the following cost function:

$$C_{11}(\tilde{\boldsymbol{\eta}}) = \left\| \bar{\mathbf{Y}} - \sum_{l=1}^{L} \tilde{\alpha}_l \bar{\mathbf{b}}(\omega_l) \right\|_2^2, \tag{72}$$

where

$$\tilde{\boldsymbol{\eta}} = \left[\mathrm{Re}^T(\tilde{\boldsymbol{\alpha}}) \ \mathrm{Im}^T(\tilde{\boldsymbol{\alpha}}) \ \boldsymbol{\omega}^T \right]^T, \tag{73}$$

with $\mathrm{Im}(\mathbf{Z})$ denotes the imaginary part of $\mathbf{Z}$, and

$$\tilde{\boldsymbol{\alpha}} = \left[\tilde{\alpha}_1 \ \tilde{\alpha}_2 \ \cdots \ \tilde{\alpha}_L \right]^T. \tag{74}$$

Denote

$$\hat{\boldsymbol{\eta}} = \arg\min_{\boldsymbol{\eta}} \ C_{10}(\boldsymbol{\eta}), \tag{75}$$

and

$$\hat{\tilde{\boldsymbol{\eta}}} = \arg\min_{\tilde{\boldsymbol{\eta}}} \ C_{11}(\tilde{\boldsymbol{\eta}}). \tag{76}$$

Let

$$f(\boldsymbol{\eta}) = \mathbf{F}\boldsymbol{\eta}, \tag{77}$$

where

$$\mathbf{F} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \tag{78}$$

with $\mathbf{I}$ and $\mathbf{0}$ denote the $L \times L$ identity matrix and the $L \times L$ matrix with zero elements, respectively. Using the EXIP principle (Söderström & Stoica a, 1989; Söderström & Stoica b, 1989), we can obtain a new estimate $\hat{\boldsymbol{\eta}}$ from $\hat{\tilde{\boldsymbol{\eta}}}$ by solving the following weighted least squares

problem

$$\hat{\hat{\boldsymbol{\eta}}} = \arg\min_{\boldsymbol{\eta}} \left[\hat{\hat{\boldsymbol{\eta}}} - f(\boldsymbol{\eta})\right]^T \mathbf{W}_{\mathrm{EXIP}} \left[\hat{\hat{\boldsymbol{\eta}}} - f(\boldsymbol{\eta})\right], \tag{79}$$

where

$$\mathbf{W}_{\mathrm{EXIP}} = E\left\{\frac{\partial^2 \left[C_{\tilde{\boldsymbol{\eta}}}(\tilde{\boldsymbol{\eta}})\right]}{\partial\tilde{\boldsymbol{\eta}}\partial\tilde{\boldsymbol{\eta}}^T}\right\}\Bigg|_{\tilde{\boldsymbol{\eta}}=\hat{\hat{\boldsymbol{\eta}}}}. \tag{80}$$

It has been shown in (Söderström & Stoica a, 1989; Söderström & Stoica b, 1989) that $\hat{\hat{\boldsymbol{\eta}}}$ is asymptotically (for large N or high SNR) statistically equivalent to $\hat{\boldsymbol{\eta}}$. The weighting matrix $\mathbf{W}_{\mathrm{EXIP}}$ is simply the *Fisher Information Matrix* (possibly scaled by a constant) for the complex-valued $\{\tilde{\alpha}_l\}_{l=1}^L$ with $\tilde{\boldsymbol{\eta}}$ replaced by its estimate $\hat{\hat{\boldsymbol{\eta}}}$. It can be easily shown that

$$\hat{\hat{\boldsymbol{\eta}}} = \left(\mathbf{F}^T\mathbf{W}_{\mathrm{EXIP}}\mathbf{F}\right)^{-1}\left(\mathbf{F}^T\mathbf{W}_{\mathrm{EXIP}}\right)\hat{\hat{\boldsymbol{\eta}}}. \tag{81}$$

The EXIP-WRELAX algorithm is composed of two steps. The first step is the same as Step 1 of the Hybrid-WRELAX algorithm and the second step is to refine the initial conditions obtained in Step 1 by using (81). Compared to the Hybrid-WRELAX algorithm, the second step of the EXIP-WRELAX algorithm is non-iterative and avoids dealing with the highly oscillatory true NLS cost function entirely. Our numerical examples show that at low SNR, the former tends to outperform the latter.

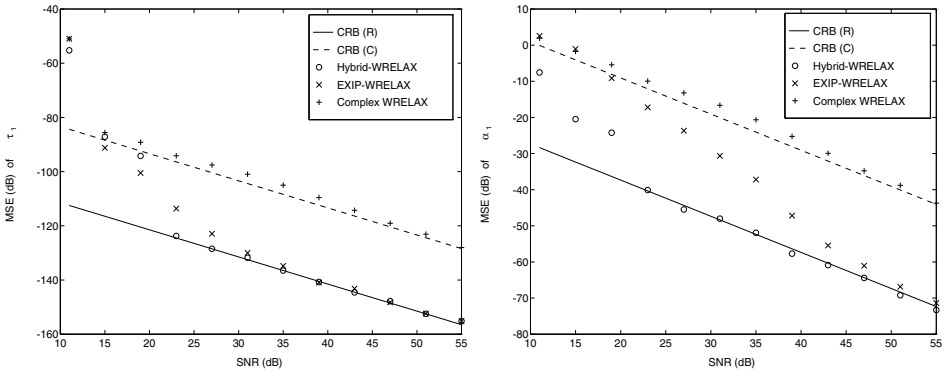


Fig. 5. Comparison of the MSEs of the WRELAX for assuming complex-valued signal amplitudes ("+"), Hybrid-WRELAX ("$\circ$"), and EXIP-WRELAX ("$\times$") with the CRBs corresponding to complex-valued (dashed line) and real-valued (solid line) signal amplitudes for (a) τ_1, (b) α_1.

The MSEs of the WRELAX ("+") for assuming $\{\alpha_l\}_{l=1}^L$ being complex-valued, Hybrid-WRELAX ("$\circ$"), and EXIP-WRELAX ("$\times$") are compared with the CRBs obtained by assuming $\{\alpha_l\}_{l=1}^L$ being complex-valued (dashed line) and real-valued (solid line) in Figure 5. Note that both Hybrid-WRELAX and EXIP-WRELAX achieve the corresponding CRB. Note also that the threshold effect is obvious in Figure 5, where the MSEs deviate away from the CRBs at low SNR. Although the WRELAX for assuming $\{\alpha_l\}_{l=1}^L$ being complex-valued also attains its corresponding CRB (dashed line) at high SNR, this wrong CRB can be larger than the true CRB by approximately 30 dB. (Note that the former CRB is expected to be

worse than the latter CRB due to the parsimony principle In this example, Hybrid-WRELAX outperforms EXIP-WRELAX at low SNR.

3.4 *M*ethod *O*f *D*irection *E*stimation based WRELAX (MODE-WRELAX) algorithm

WRELAX was extended in to deal with the real-valued signals with highly oscillatory correlation functions (Hybrid-WRELAX and EXIP-WRELAX). The resolution of WRELAX are much higher than that of the conventional matched filter approach. However, when the signals are very closely spaced in arrival times, the convergence speed of WRELAX decreases rapidly. Here, we study how MODE can be used with our efficient WRELAX algorithm for super resolution time delay estimation. The new algorithm is referred to as MODE-WRELAX. Although MODE can provide very poor amplitude estimates and WRELAX has the slow convergence problem, MODE-WRELAX outperforms both MODE and WRELAX. MODE-WRELAX can be used for both complex- and real-valued signals (including those with highly oscillatory correlation functions).

The same data model as that in the Section 2 is adopted. Here, the transmitted signal $s(t)$ represents an arbitrary known transmitted signal. We assume that $s(t)$, $y(t)$, $e(t)$ and $\{\alpha_l\}_{l=1}^{L}$ are either all complex-valued or all real valued, which will be dealt with in the following respectively.

3.4.1 MODE-WRELAX for complex-valued signals

Assume that $(\cdot)^T$ denote the transpose and let

$$\mathbf{A} = \left[\mathbf{a}(\omega_1)\ \mathbf{a}(\omega_2)\ \cdots\ \mathbf{a}(\omega_L) \right]^T, \tag{82}$$

where $\mathbf{a}(\omega_l)$ is same as that in (10).

Then the data model (4) can be written in the following vector form:

$$\mathbf{Y} = \mathbf{S}\mathbf{A}\boldsymbol{\alpha} + \mathbf{E}. \tag{83}$$

where $\mathbf{Y}$, $\mathbf{S}$ and $\mathbf{E}$ are same as that in (7)-(9), $\boldsymbol{\alpha}$ is same as that in (70).

When $\mathbf{S}$ is an identity matrix, then the above time delay estimation issue becomes a sinusoidal parameter estimation problem and MODE is an asymptotically statistically efficient estimator of $\{\omega_l\}_{l=1}^{L}$ for complex-valued signals (Stoica & Sharman a, 1990; Stoica & Sharman b, 1990). The MODE algorithm (Stoica & Sharman a, 1990; Stoica & Sharman b, 1990) can be easily extended to the data model in (83) where $\mathbf{S}$ is an arbitrary diagonal matrix as follows. The MODE estimates $\{\hat{\omega}_l\}_{l=1}^{L}$ of $\{\omega_l\}_{l=1}^{L}$ can be obtained by minimizing the following cost function

$$C_{12}(\{\omega_l\}_{l=1}^{L}) = \mathbf{Y}^H \mathbf{P}_{\tilde{\mathbf{A}}}^{\perp} \mathbf{Y}, \tag{84}$$

and

$$\mathbf{P}_{\tilde{\mathbf{A}}}^{\perp} = \mathbf{I} - \tilde{\mathbf{A}} \left(\tilde{\mathbf{A}}^H \tilde{\mathbf{A}} \right)^{-1} \tilde{\mathbf{A}}^H, \tag{85}$$

$$\tilde{\mathbf{A}} = \mathbf{S}\mathbf{A}. \tag{86}$$

To avoid the search over the parameter space, $C_{10}(\{\omega_l\}_{l=1}^{L})$ can also be reparametrized in terms of another parameter vector $\tilde{\mathbf{b}} = \left[\tilde{b}_0\ \tilde{b}_1\ \cdots\ \tilde{b}_L \right]^T$, where $\{\tilde{b}_l\}_{l=0}^{L}$ are the coefficients of

the following polynomial:

$$\tilde{b}(z) \triangleq \sum_{l=0}^{L} \tilde{b}_l z^{L-l} \triangleq \tilde{b}_0 \prod_{l=1}^{L} (z - e^{j\omega_l});\ \tilde{b}_0 \neq 0. \tag{87}$$

Since the polynomial $\tilde{b}(z)$ in (87) has all of its zeros on the unit circle, its coefficients $\{\tilde{b}_l\}$ satisfy the conjugate symmetry constraint (Stoica & Sharman a, 1990):

$$\tilde{b}_l = \tilde{b}_{L-l}^*, \quad l = 0, 1, \cdots, L, \tag{88}$$

where $(\cdot)^*$ denotes the complex conjugate. Let

$$\mathbf{B} = \begin{bmatrix} \tilde{b}_0 & & 0 \\ \vdots & \ddots & \\ \tilde{b}_L & & \tilde{b}_0 \\ & \ddots & \vdots \\ 0 & & \tilde{b}_L \end{bmatrix} \in \mathcal{C}^{N \times (N-L)}. \tag{89}$$

Assume that the diagonal elements of $\mathbf{S}$ are nonzero (see **Remark 1** for more discussions). Let

$$\tilde{\mathbf{B}} = \mathbf{S}^{-H}\mathbf{B}. \tag{90}$$

It can be readily verified that $\mathbf{B}^H\mathbf{A} = \mathbf{0}$ and hence $\tilde{\mathbf{B}}^H\tilde{\mathbf{A}} = \mathbf{0}$. Then $\mathbf{P}_{\tilde{\mathbf{A}}}^{\perp} = \tilde{\mathbf{B}}\left(\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\right)^{-1}\tilde{\mathbf{B}}^H$ and minimizing $C_{12}(\{\omega_l\}_{l=1}^L)$ in (84) is equivalent to minimizing

$$C_{13}(\{b_l\}_{l=0}^L) = \mathbf{Y}^H\tilde{\mathbf{B}}\left(\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}\right)^{-1}\tilde{\mathbf{B}}^H\mathbf{Y}. \tag{91}$$

Note that $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}$ in (91) can be replaced by a consistent estimate without affecting the asymptotically statistical efficiency of the minimizer of (91). Hence $\hat{\mathbf{b}}$ can be obtained computationally efficiently as follows:

$$\hat{\tilde{\mathbf{b}}} = \arg\min_{\tilde{\mathbf{b}}} \left[\mathbf{Y}^H\mathbf{S}^{-H}\mathbf{B}\left(\hat{\mathbf{B}}_0^H\mathbf{S}^{-1}\mathbf{S}^{-H}\hat{\mathbf{B}}_0\right)^{-1}\mathbf{B}^H\mathbf{S}^{-1}\mathbf{Y}\right], \tag{92}$$

where $\hat{\mathbf{B}}_0$ is the initial estimate of $\mathbf{B}$ obtained by replacing $\tilde{\mathbf{b}}$ with $\hat{\tilde{\mathbf{b}}}^{(0)}$ in (89). The initial value $\hat{\tilde{\mathbf{b}}}^{(0)}$ is obtained by setting $\tilde{\mathbf{B}}^H\tilde{\mathbf{B}}$ in (91) to $\mathbf{I}$:

$$\hat{\tilde{\mathbf{b}}}^{(0)} = \arg\min_{\tilde{\mathbf{b}}} \left[\mathbf{Y}^H\mathbf{S}^{-H}\mathbf{B}\mathbf{B}^H\mathbf{S}^{-1}\mathbf{Y}\right]. \tag{93}$$

To avoid the trivial solution $\tilde{\mathbf{b}} = \mathbf{0}$, we should impose $\| \tilde{\mathbf{b}} \| = 1$ in (92) and (93) or some other similar constraints. (For detailed implementation steps, see Section 4.) The estimates $\{\hat{\omega}_l\}_{l=1}^L$ of $\{\omega_l\}_{l=1}^L$ are the phases of the roots of the polynomial $\sum_{l=0}^L \hat{\tilde{\mathbf{b}}}_l z^{L-l}$. Once $\{\hat{\omega}_l\}_{l=1}^L$ are obtained, the amplitudes $\boldsymbol{\alpha}$ are estimated by applying the linear least-squares approach to

$$\mathbf{Y} \approx \mathbf{S}\hat{\mathbf{A}}\boldsymbol{\alpha}, \tag{94}$$

where $\hat{\mathbf{A}}$ is formed by replacing $\{\omega_l\}_{l=1}^L$ with $\{\hat{\omega}_l\}_{l=1}^L$ in (82).

Remark 1: MODE cannot be implemented efficiently to avoid the search over the parameter space when $S(k) = 0$ for some k. The most commonly used complex analytic signal $s(t)$ is low-pass. For this case, we can select a contiguous segment of $\mathbf{Y}$ satisfying $|S(k)| > 0, \quad K_1 \leq k \leq K_2$, and preferrably with $|S(k)|$ above a certain threshold to avoid numerical problems. We can then apply MODE to the segment $\{Y(k)\}_{k=K_1}^{K_2}$ to estimate $\{\omega_l\}_{l=1}^L$.

Remark 2: The amplitude estimates given above can be very poor when the SNR is not sufficiently high. This is because some of the MODE estimates $\{\hat{\omega}_l\}_{l=1}^L$ can be so closely spaced that $\hat{\mathbf{A}}$ in (94) is seriously ill-conditioned. We use a simple spacing adjustment scheme to avoid this problem. After obtaining the MODE estimates $\{\hat{\omega}_l\}_{l=1}^L$ of $\{\omega_l\}_{l=1}^L$, we first sort them in the ascending order and then check the spacing between two adjacent estimates. If the distance between any two estimates, say $\hat{\omega}_1$ and $\hat{\omega}_2$ ($\hat{\omega}_1 \leq \hat{\omega}_2$), is smaller than a predefined threshold, say $\triangle\omega_t$, we adjust the estimates by replacing $\hat{\omega}_1$ with $\hat{\omega}_1 - 0.5\triangle\omega_t$ and $\hat{\omega}_2$ with $\hat{\omega}_2 + 0.5\triangle\omega_t$. The amplitudes are then estimated using the adjusted estimates of $\{\omega_l\}_{l=1}^L$. This spacing adjustment step is *ad hoc* but can be used to provide good initial delay and amplitude estimates to replace the first $L-1$ steps of WRELAX.

The MODE estimates $\{\hat{\omega}_l\}_{l=1}^L$ of $\{\omega_l\}_{l=1}^L$ and $\{\hat{\alpha}_l\}_{l=1}^L$ of $\{\alpha_l\}_{l=1}^L$, which may not be optimal, especially for real-valued signals, can be refined by using the last step of the WRELAX algorithm.

WRELAX is a relaxation-based minimizer of the following nonlinear least-squares (NLS) criterion:

$$C_{14}(\{\alpha_l, \omega_l\}_{l=1}^L) = \| \mathbf{Y} - \sum_{l=1}^{L} \alpha_l \mathbf{S}\mathbf{a}(\omega_l) \|^2 . \tag{95}$$

When the signals are not spaced very closely, WRELAX usually converges in a few steps. However, when the signals are very closely spaced, the convergence speed of WRELAX is very slow. Yet by using the above MODE algorithm to obtain the initial conditions and then using the last step of WRELAX to refine them, super resolution time delay estimation can be achieved with a fast convergence speed.

Before we present the MODE-WRELAX algorithm, let us consider the following preparations. Let

$$\mathbf{Y}_l = \mathbf{Y} - \sum_{i=1, i\neq l}^{L} \hat{\alpha}_i [\mathbf{S}\mathbf{a}(\hat{\omega}_i)], \tag{96}$$

where $\{\hat{\alpha}_i, \hat{\omega}_i\}_{i=1, i\neq l}^L$ are assumed to be given . Then (95) becomes

$$C_{15}(\alpha_l, \omega_l) = \| \mathbf{Y}_l - \alpha_l \mathbf{S}\mathbf{a}(\omega_l) \|^2 . \tag{97}$$

Minimizing $C_{15}(\alpha_l, \omega_l)$ with respect to ω_l and the complex-valued α_l yields

$$\hat{\omega}_l = \arg\max_{\omega_l} \left| \mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l) \right|^2 , \tag{98}$$

and

$$\hat{\alpha}_l = \left. \frac{\mathbf{a}^H(\omega_l)(\mathbf{S}^*\mathbf{Y}_l)}{\| \mathbf{S} \|_F^2} \right|_{\omega_l = \hat{\omega}_l} . \tag{99}$$

With the above preparations, we now present the steps of the MODE-WRELAX algorithm for complex-valued signals.

Step (1): Select a contiguous segment of data vector $\mathbf{Y}$ (for MODE use only) so that $|S(k)| > 0, \quad K_1 \leq k \leq K_2$. Apply MODE to the segment to obtain $\{\hat{\omega}_l\}_{l=1}^{L}$. Adjust $\{\hat{\omega}_l\}_{l=1}^{L}$ so that the minimum spacing of $\{\hat{\omega}_l\}_{l=1}^{L}$ is at least $\triangle\omega_t$. Obtain the estimates $\{\hat{\alpha}_l\}_{l=1}^{L}$ of $\{\alpha_l\}_{l=1}^{L}$ by using (94).

Step (2): Refine the estimates obtained in Step (1) by using the last step of WRELAX. That is, compute $\mathbf{Y}_1$ by using $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}^{L}$ obtained in Step (1). Obtain $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1}$ from $\mathbf{Y}_1$ by using (98) and (99). Next, compute $\mathbf{Y}_2$ by using the updated $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1,3,\cdots,L}$ and determine $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=2}$ from $\mathbf{Y}_2$. Then compute $\mathbf{Y}_3$ by using the updated $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=1,2,4,\cdots,L}$ and determine $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=3}$ from $\mathbf{Y}_3$. Continue this procedure and similarly determine $\{\hat{\omega}_l, \hat{\alpha}_l\}_{l=L}$ from $\mathbf{Y}_L$. Repeat the above process until "practical convergence".

Similarly, we can use MODE as an initialization method for the EM time delay estimation algorithm (Moon, 1996), which is referred to as MODE-EM. However, we have found through numerical simulations that the convergence speed of MODE-EM is slower than that of MODE-WRELAX.

3.4.2 MODE-WRELAX for real-valued signals

For bandpass real-valued signals, the cost function in (95) is a highly oscillatory cost function and is very difficult to find its global minimum. Although MODE is derived for complex-valued signals, we can apply it by assuming the real-valued amplitudes $\{\alpha_l\}_{l=1}^{L}$ to be complex-valued. These initial estimates are then refined by the WRELAX algorithm. Since the attraction domain of the cost function (95) is extremely small, a very good initial condition is required to achieve the global convergence of any minimizer of (95). The MODE estimates are first refined by WRELAX by assuming $\{\alpha_l\}_{l=1}^{L}$ to be complex-valued since the attraction domain of (95) becomes much larger when assuming the real-valued $\{\alpha_l\}_{l=1}^{L}$ to be complex-valued (Manickam et al., 1994). The so-obtained estimates are refined again by WRELAX by using the fact that $\{\alpha_l\}_{l=1}^{L}$ are real-valued.

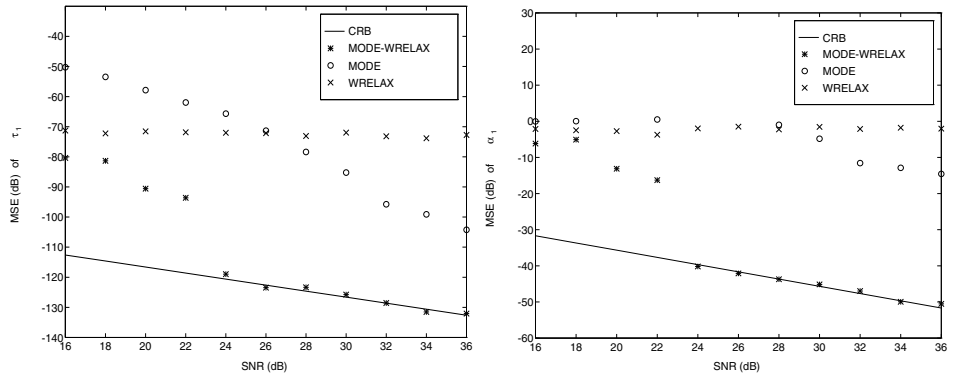
With the above preparations, we now present the steps of the MODE-WRELAX algorithm for real-valued signals.

Step (1): Select a contiguous segment of data vector $\mathbf{Y}$ so that $|S(k)| > 0, \quad K_1 \leq k \leq K_2$. By assuming the real-valued $\{\alpha_l\}_{l=1}^{L}$ to be complex-valued, obtain the estimates $\{\hat{\omega}_l\}_{l=1}^{L}$ and $\{\hat{\alpha}_l\}_{l=1}^{L}$ in the same way as Step (1) of the MODE-WRELAX algorithm for complex-valued signals.

Step (2): Refine the estimates obtained in Step (1) above by using the last step of WRELAX by assuming complex-valued signals. Take the real parts of the so-obtained amplitude estimates as the amplitude estimates $\{\hat{\alpha}_l\}_{l=1}^{L}$ of $\{\alpha_l\}_{l=1}^{L}$.

Step (3): Refine the estimates obtained in Step (2) above by using the last step of WRELAX and the fact that the signals are real-valued.

Now we consider an example where $\tau_2 - \tau_1 = 0.2\tau_e$. The MSEs of MODE ("$\circ$"), WRELAX ("$\times$"), and MODE-WRELAX ("$*$") are compared with the corresponding CRBs (solid line) in Figure 6. Note that due to the highly oscillatory cost functions and very closely spaced signals, WRELAX converges to some local minimum instead of the global one, which yields



(a) (b)

Fig. 6. Comparison of the MSEs of WRELAX ("×"), MODE("○"), and MODE-WRELAX ("∗") with the CRBs (solid line) corresponding to real-valued signals for (a) τ_1, (b) α_1.

very poor estimates. Since the MODE amplitude estimates are obtained without spacing adjustment, they are so poor at low SNR that some of their MSEs are above the axis limit due to the inversion of ill-conditioned matrices corresponding to very closely spaced delay estimates. Although the MSEs of the MODE estimates are close to the CRBs corresponding to the complex-valued amplitudes when the SNR is high, the wrong CRBs (not shown to avoid too many lines in the figure) can be larger than the true CRBs, which correspond to the real-valued amplitudes, by approximately 30 dB.

4. Applications

4.1 Pavement profiling via Ground Penetrating Radar (GPR)

The detection and classification of roadway subsurface anomalies are very important for the design and quality evaluation of highways. Ultra wideband ground penetrating radar emits nonsinusoidal impulses with extremely large bandwidth (several GHz) and is very suitable for this application because of its high range resolution (on the order of several centimeters). The returned echoes of the ultra wideband ground penetrating radar are superimposed real-valued signals reflected from the boundaries of different media (layers, voids, etc.), which can be described by (1). Both the delays and gains are very useful for the detection and classification of roadway subsurface anomalies. The delays can be used to determine the layer thickness or anomaly location and the gains can be used to classify the type of media because the gains are related to the reflection coefficient at the boundary between two media with different dielectric constants. Once we get the estimates of the media dielectric constants, we can judge the type of the media.

WRELAX has been applied to layer stripping, based on a single-scan basis using experimental data and has been proven to be very useful. However, the straightforward extension of WRELAX from one scan to multi-scans is not feasible since, under moving conditions, the antenna can change its height relative to the road surface. As the height above the road decreases (or increases), the amplitude and the time delay of the received signal changes as well, which can lead to incorrect estimates of layer thickness and permittivity. A motion compensation and parameter estimation algorithm for GPR is presented for simultaneous

motion compensation and time delay estimation. More details can be found in our papers (Li & Wu, 1998; Su & Wu, 2000; Wu & Li, 1998; Wu et al., 1999; 2002).

4.2 Acoustic proximity ranging system

Proximity ranging is very important in a wide range of remote sensing applications, including level detection, robot manipulation, process control, nondestructive testing, and cavity thickness monitoring. Various types of sensors based on different physical principles such as capacitive or inductive proximity sensors, laser displacement sensors, and ultrasonic sensors, can be used to perform proximity ranging. Among these sensors, ultrasonic sensors have many important advantages over the others. However, due to the presence of secondary echoes, using ultrasonic sensors for very close proximity ranging is very difficult. When ultrasonic sensors face a sound-hard reflective surface, the reflected sound wave can bounce back and forth several times between the sensors and the reflection surface before decaying to zero, which results in unwanted strong and overlapping secondary echoes in the received signal. The time delays for the secondary echoes are approximately integer multiples of the time delay of the first echo. The matched filter based methods cannot resolve two echoes with a time spacing less than the reciprocal of the signal bandwidth. Hence, for most of the very short distance measurement scenarios, the matched filter based algorithms tend to fail or suffer from severe performance degradations due to their poor resolutions. WRELAX and its extended version as a super-resolution time delay estimation approaches are developed for general purposes. They do not exploit the a *priori* information of the integer multiple time delays and the nonnegative amplitude due to the acoustically hard reflections. More details can be found in our papers (Li et al. a, 2002; 2003; Li et al. b, 2003).

5. Conclusion

A family of relaxation- and FFT-based efficient time-delay estimation algorithms were presented for different scenarios in this chapter. By avoiding the computationally demanding multidimensional search over the parameter space, the proposed algorithms minimize the NLS criterion at a much lower implementation cost. They are more efficient and systematic than existing algorithms. Some practical applications utilizing the proposed algorithms are also presented. Theoretical analysis and simulations demonstrate the efficiency of the proposed new algorithms.

6. Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (No. 60879019, No. 61172112, No. 61179064) and in part by the Foundation of Civil Aviation Administration of China (MHRD0606).

7. References

Barton, D. K.(1988). Modern Radar System Analysis Artech House Inc..

Bell, B. M. & Ewart, T. E. (1986). Separating multipaths by global optimization of multidimensional matched filter, IEEE Transactions on Acoustic, Speech and Signal Processing 34(5): 1029-1037.

Bian, Y. & Last, D. (1997). Eigen-decomposition techniques for Loran-C skywave estimation, IEEE Transactions on Aerospace and Electronic Systems 33(1): 117-124.

Blackowiak, A. D. & Rajan, S. D. (1995). Multipath arrival estimates using simulated annealing: Application to crosshole tomography experiment, IEEE Journal of Oceanic Engineering 20(3): 157-165.

Bruckstein, A. M., Shan, T. J. & Kailath, T. (1985). The resolution of overlapping echoes, IEEE Transactions on Acoustics, Speech, and SignalProcessing 33(6): 1357-1367.

Ehrenberg, J. E., Ewatt, T. E. & Morris, R. D. (1978). Signal processing techniques for resolving individual pulses in multipath signal, Journal of the Acoustics Society of America 63(6): 1861-1865.

Feder, M. & Weinstein, E. (1988). Parameter estimation of superimposed signals using the EM algorithm, IEEE Transactions on Acoustic, Speech and Signal Processing 36(4): 477-489.

Gedalyahu, K. & Eldar, Y. (2010). Time-delay estimation from low-rate samples: A union of subspaces approach, IEEE Transactions on Signal Processing 58(6): 3017-3031.

He, W., Wu, R. & Liu, J. (2011). Identification Method of EM Property Inversion for Multilayer Media, IEEE Radar Conference, 2011.

Karmanov, V. G.(1977). Programmation Mathematique, Editions Mir, Moscow.

Kay, S. M.(1988). Modern Spectral Estimation: Theory and Application. Prentice-Hall.

Kirsteins, I. P. (1987). High resolution time delay estimation, Proceedings of ICASSP 87 pp. 451-454.

Kirsteins, I. P. & Kot, A. C. (1990). Performance analysis of a high resolution time delay estimation algorithm, Proceedings of ICASSP'90 pp. 2767-2770.

Li, J. & Stoica, P.(1996). Efficient mixed-spectrum estimation with applications to target feature extraction, IEEE Transactions on Signal Processing 44(2), 281-295.

Li, J. & Wu, R. (1998). An efficient algorithm for time delay estimation, IEEE Transactions on Signal Processing 46(8): 2231-2235.

Li, X., Wu, R., Sheplak, M., & Li, J. (2002). Multifrequency CW-based Time-Delay Estimation for Proximity Ultrasonic Sensors, IEE Proc.-Radar Sonar Navig 149(2): 53-59.

Li, J., Halder, B., Stoica, P. & Viberg, M.(1995). Decoupled Maximum Likelihood Angle Estimationfor Signals with Known Waveforms, IEEE Transactions on Signal Processing 43(9): 2154-2163.

Li, X., Wu, R., Rasmi, S. , Li, J. , Cattafesta III, L. N. , & Sheplak M. (2003). Acoustic Proximity Ranging in the Presence of Secondary Echoes, IEEE Transactions on Instrumentation and Measurement 52(5): 1593-1605.

Li, X., Wu, R., Rasmi, S., Li, J., Cattafesta III, L. N. & Sheplak, M. (2003). An Acoustic Proximity Ranging System for Monitoring the Cavity Thickness, IEEE Transactions on ultrasonics, ferroelectrics, and frequency control 50(7): 898-910.

Liu, N., Xu, Z. & Sadler, B. (2010). Ziv-zakai time-delay estimation bounds for frequency-hopping waveforms under frequency-selective fading, IEEE Transactions on Signal Processing 58(12): 6400-6406.

Manickam, T. G., Vaccaro, R. J. & Tufts, D. W. (1994). A least-squares algorithm for multipath time-delay estimation, IEEE Transactions on Signal Processing 42(11): 3229-3233.

Manickam, T. G., Vaccaro, R. J. & Tufts, D. W.(1994). A least-squares algorithm for multipath time-delay estimation, IEEE Transactions on Signal Processing, 42(11): 3229-3233.

Moon, T. K. (1996). The expectation-maximization algorithm, IEEE Signal Processing Magazine pp. 47-60.

Roy, R., Paulraj, A. & Kailath, T.(1986). ESPRIT - A Subspace Rotation Approach to Estimation of Parameters of Cisoids in Noise. IEEE Transactions on Acoustics, Speech, and Signal Processing 34(5), 1340-1342.

Schmidt, R. O. (1986). Multiple emitter location and signal parameter estimation, IEEE Transactions on Antennas and Propagation 34(3): 276-280.

Söderström, T. & Stoica, P.(1989). System Identification, Prentice-Hall International.

Stoica, P. & Nehorai, A. (1990). Performance study of conditional and unconditional direction-of-arrival estimation, IEEE Transactions on Acoustics, Speech, and Signal Processing 38(10): 1783-1795.

Stoica, P. & Söderström, T. (1989). On Reparametrization of Loss Functions Used in Estimation and the Invariance Principle, Signal Processing 17(4): 383-387.

Stoica, P. & Sharman, K. C.(1990). Novel Eigenanalysis Method for Direction Estimation, IEE Proceedings, Radar and Signal Processing F 137(1): 19-26.

Stoica, P. & Sharman, K. C.(1990). Maximum Likelihood Methods for Direction-of-Arrival Estimation, IEEE Transactions on Acoustics, Speech, and Signal Processing, 38(7): 1132-1143.

Stoica, P. & Moses, R. L.(1997). Introduction to Spectral Analysis. Prentice-Hall.

Stewart, G. W.(1973). Introduction to Matrix Computations Academic Press, Inc..

Su, Z. & Wu, R. (2000). Delay and doppler scale estimation of multiple moving targets via ds-wrelax, Electronics Letters 36(9): 827-828.

Tufts, D. W., Kumaresan, R. & Kirsteins, I. (1982). Data adaptive signal estimation by singular value decomposition of a data matrix, Proceedings of the IEEE 70: 684-685.

Vaccaro, R. J., Ramalingam, C. S. & Tufts, D. W.(1992). Least-squares time-delay estimation for transient signals in a multipath environment, Journal of Acoustical Society of America 92: 210-218.

Wang, Z., Li, J. & Wu, R. (2005). Time-delay-and time-reversal-based robust Capon beamformers for ultrasound imaging, IEEE Transactions on Medical Imaging 24(10): 1308-1322.

Wu, R. & Li, J. (1998). Time-delay estimation via optimizing highly oscillatory cost functions, IEEE Journal of Oceanic Engineering 23(3): 235-244.

Wu, R. & Li, J. (1999). Time delay estimation with multiple looks in colored gaussian noise, IEEE Transactions on Aerospace and Electronic Systems 35(4): 1354-1361.

Wu, R., Li, J. & Liu, Z. (1999). Super resolution time delay estimation via mode-wrelax, IEEE Transactions on Aerospace and Electronic Systems 35(1): 294-307.

Wu, R., Li, X. & Li, J. (2002). Continuous pavement profiling with ground-penetrating radar, IEE Proceedings-Radar, Sonar and Navigation, 149(4): 183-193.

Wu, R., Cao, Y. & Liu, J. (2010). Multilayered Diffraction Tomography Algorithm for Ground Penetrating Radar, International Conference on Signal Processing, pp.2129-2132.

Zangwill, W. I.(1967). Nonlinear Programming: A Unified Approach Prentice-Hall, Inc., Englewood Cliffs, N.J..

Zehna, P. W.(1966). Invariance of maximimum likelihood estimation. Annual Mathematic Statistics 37:755.

4

Channel Identification for OFDM Communication System in Frequency Domain

Lianming Sun
The University of Kitakyushu
Japan

1. Introduction

Orthogonal frequency division multiplexing (OFDM) modulation has excellent performances, for example, strong tolerance against multipath interferences, effective spectral efficiency, high information capacity and simplicity of equalization. Consequently, it has been widely utilized in the services of digital terrestrial broadcasting, asymmetric digital subscriber line (ADSL), local wireless LAN and optical fiber communications. In the transmitter, relay station and receiver, signal processing techniques are used to mitigate the effects caused by various interferences, carrier frequency offset and noise, then to improve the equalization precision of information data. These techniques may achieve the utmost of their effectiveness if the reliable knowledge of the communication channel is applicable. Nevertheless, the prior information of the OFDM channel dynamics is typically unavailable, whereas the practical channel is often time-varying due to the differing propagation paths, scattering and reflection of electric waves. Hence it is necessary to identify the channel model from the observation data and some distinctive structural information inserted in the OFDM signals. In this chapter some channel identification problems as well as the fundamental mathematical tools are discussed, and several frequency domain algorithms are investigated.

Channel information is an essential issue in practical communication systems. It is often obtained by channel identification, which may be performed either in the time domain or in the frequency domain (Giannakis et al., 2000). The identification algorithms in time domain are commonly executed through least mean square (LMS) method, recursive least squares (RLS) method, maximum likelihood (ML) when a known sequence of training symbols transmitted in some specified training styles (Haykin, 2001; Ljung, 1999). When no training sequence can be used for channel identification, the blind (Chi et al., 2006; Ding & Li, 2001) or semi-blind algorithms may use some statical or structured properties of the OFDM signals, for example, the cyclic prefix, the symbol pattern of constellation (Koiveunen et al., 2004). If the spatial information is available, the subspace method is the possible choice (Muquet et al., 2002). These algorithms have been utilized in channel estimation and equalization, and have helped to improve the communication performance in applications of equalization (Giannakis et al., 2000), compensation of frequency offset (Yu & Su, 2004), compensation of nonlinearity distortion (Ding et al., 2004), interference compensation in relay station (Shibuya, 2006; Sun & Sano, 2005).

Nevertheless, in the presence of multipath interferences with long tags, or with the severe restriction that the carriers outside the signal band width do not convey any information

symbols, the time domain algorithms may suffer from either low convergence rate or high computational complexity. Furthermore, if few training sequences can be available for channel estimation, the blind algorithms in the time domain commonly have to employ nonlinear optimization which may converge slowly.

On the other hand, the OFDM signals in base band are managed by Fourier transform and inverse Fourier transform, it implies that the channel identification can also be performed in the frequency domain with the aid of Fourier transform. The advantages of OFDM channel identification in the frequency domain are as follows: Both the transmitted and received signals in base band, and the dynamic channel model can be treated conveniently through Fourier transform in the frequency domain, while fast Fourier transform (FFT) and inverse fast Fourier transform (IFFT) can significantly reduce the computational complexity in channel identification. Additionally, the dynamics of channel model is easily handled in frequency domain without extra computation even for long delay taps, and only simple computation is required for convolution and deconvolution. Furthermore, the scattered pilot symbols assigned at some specified carriers can be more applicable than that in the time domain, and the identification algorithm can easily be combined with equalization, interference cancellation. Hence it is a strong motivation to develop effective channel identification algorithms in the frequency domain.

In this chapter the channel identification is studied in the frequency domain, and several identification algorithms are presented for the OFDM channel working under severe communication environment or restricted identification conditions. Firstly, the frequency properties of both the OFDM signals in base band and the propagation channel used in identification are briefly illustrated, and some structural features of cyclic prefix, constellation of information symbols and scattered pilot symbols are also shown in the frequency domain. Secondly, the fundamentals of identification algorithms are discussed, including the frequency properties of the inter-symbol interference (ISI) and inter carrier interference (ICI), the correlation function and spectral property of various signals in OFDM system, the leakage error of Fourier transform. Then, several identification algorithms are presented, including the batch processing algorithm, recursive algorithm, the usage of pilot symbols, the method to mitigate the affection of equalization errors for the case of low pilot rate. Next, the applications of the identification algorithms are considered for the cases where the multipath interferences have long delay taps, the OFDM signal has severe bandwidth restriction, or the propagation channel has fast fading. Furthermore, their performances of convergence and computational complexity are analyzed, and compared with the methods in the time domain. It is seen that Fourier transform is a powerful mathematical tool in the identification problems of OFDM channel, and the Fourier transform based algorithms demonstrate attractive performance even under some severe communication conditions.

2. Fundamentals in channel identification

2.1 Guard interval

Let the normalized period of OFDM information symbol be denoted as N. As shown in Fig.1, OFDM guard interval (GI) attaches a copy of the effective symbol's tail part to its head as a cyclic prefix when the signal is transmitted. Let the GI length be N_{gi}, then the practical transmission period denoted as N_{tx} becomes to $N_{\mathrm{tx}} = N + N_{\mathrm{gi}}$.

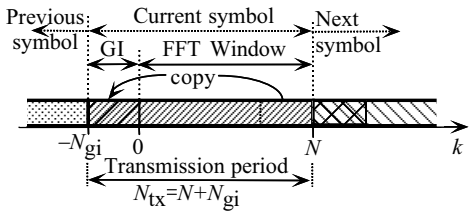


Fig. 1. Guard interval in OFDM signal

2.2 OFDM signals in base band

In the transmission symbol period N_{tx}, the transmitted signal in base band is generated by N-point IFFT as follows

$$d(k)=\sum_{n=-N/2+1}^{N/2} D(n,l)e^{jn\omega_0(k-lN_{tx})},\ \text{for}\ lN_{tx}-N_{gi}\le k<lN_{tx}+N, \tag{1}$$

where the FFT size N is a power of 2. k and $\omega_0=2\pi/N$ are the normalized sampling instant and angular frequency factor, respectively. $D(n,l)$ is the symbol conveyed at the nth carrier in the lth transmission symbol period, and it belongs to a modulation constellation with finite elements.

2.3 Pilot and information symbols

For the purpose of synchronization and equalization, scattered pilot symbols are assigned at the specified carriers, i.e., at these pilot carriers, the transmitted symbols $D(n,l)$ are the known ones at both the transmitter and the receiver, and can be employed in channel identification as well as symbol equalization.

On the other hand, the symbol $D(n,l)$ at information carrier can generally be treated as a random sequence with respect to the carrier number n and symbol period l, i.e.,

$$\lim_{L\to\infty}\frac{1}{L}\sum_{l=1}^{L} D^*(n_1,l)D(n_2,l-l_1)=\bar{D}^2\delta(n_1-n_2)\delta(l_1) \tag{2}$$

holds true, where δ is the delta function, * denotes the conjugate complex, $\bar{D}^2$ is the mean square of the constellation, n_1 and n_2 are the carrier numbers, l_1 is an arbitrary integer.

2.4 Multipath channel model

Assume that the received signal in base band under multipath environment can be approximated by

$$y(k)=\sum_{m=0}^{M} r_m(k)+e(k)=\sum_{m=0}^{M} h_m d(k-k_m)+e(k), \tag{3}$$

where $r_m(k)$ is the mth multipath wave to the receiver, h_m is its coefficient, k_m is the delay tap, and $e(k)$ is the additive noise. Correspondingly the channel model can be expressed by z transform as

$$H(z)=h_0+h_1z^{-k_1}+h_2z^{-k_2}+\cdots+h_Mz^{-k_M}, \tag{4}$$

where z^{-1} is a backward shift operator, k_M is the longest effective delay tap of interference. Substituting $z = e^{j\omega_0}$ into (4) also yields the frequency response function of the channel model.

3. Identification of multipath channel with long delay taps

When the delay taps of multipath waves are within GI, both equalization and channel identification are easily implemented in OFDM communication (Koiveunen et al., 2004; Wang & Poor, 2003). However, the situation is quite different when the delay taps of some multipath waves exceed GI due to the induced inter-symbol interference (ISI) and inter-carrier interference (ICI) (Suzuki et al., 2002).

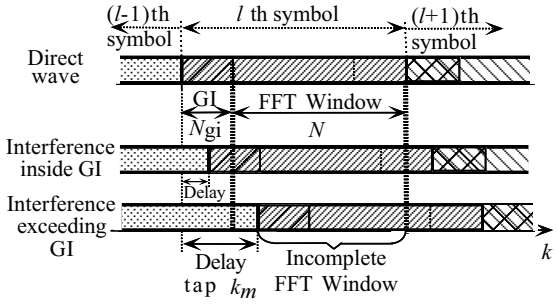


Fig. 2. Multipath interference in OFDM system

3.1 Signal properties used in identification

3.1.1 Interference exceeding GI

Consider the interference $r_m(k)$ with long delay tap k_m exceeding GI. In the lth effective symbol period for $lN_{\text{tx}} \le k < lN_{\text{tx}} + N$, the component of interference with delay tap k_m in Fig.2 is given by

$$r_m(k) = h_m d(k-k_m) = \begin{cases} h_m \sum_{n=-N/2+1}^{N/2} D(n,l-1) e^{jn\omega_0(k-k_m+N_{\text{gi}}-lN_{\text{tx}})} \\ \qquad \text{for } k-k_m-lN_{\text{tx}} < -N_{\text{gi}}, \\ h_m \sum_{n=-N/2+1}^{N/2} D(n,l) e^{jn\omega_0(k-k_m-lN_{\text{tx}})} \\ \qquad \text{for } -N_{\text{gi}} \le k-k_m-lN_{\text{tx}} < N. \end{cases} \tag{5}$$

Performing N-point FFT of $r_m(k)$ within the FFT window for $lN_{\text{tx}} \le k < lN_{\text{tx}} + N$ yields the frequency components of $r_m(k)$. For example, the component corresponding to the nth carrier is expressed by

$$\frac{1}{N} \sum_{k=lN_{\text{tx}}}^{lN_{\text{tx}}+N-1} r_m(k) e^{-jn\omega_0(k-lN_{\text{tx}})} = \frac{1}{N} \sum_{k=0}^{N-1} r_m(k+lN_{\text{tx}}) e^{-jn\omega_0 k}$$
$$= \frac{1}{N} \left(\sum_{k=0}^{k_m-N_{\text{gi}}-1} r_m(k+lN_{\text{tx}}) e^{-jn\omega_0 k} + \sum_{k=k_m-N_{\text{gi}}}^{N-1} r_m(k+lN_{\text{tx}}) e^{-jn\omega_0 k} \right)$$

$$= \frac{h_m}{N} \sum_{k=0}^{k_m - N_{\text{gi}} - 1} \left(\left(\sum_{n_1=-N/2+1}^{N/2} D(n_1, l-1) e^{jn_1\omega_0(k-k_m+N_{\text{gi}})} \right) e^{-jn\omega_0 k} \right)$$

$$+ \frac{h_m}{N} \sum_{k=k_m-N_{\text{gi}}}^{N-1} \left(\left(\sum_{n_1=-N/2+1}^{N/2} D(n_1, l) e^{jn_1\omega_0(k-k_m)} \right) e^{-jn\omega_0 k} \right)$$

$$= \frac{h_m}{N} \sum_{k=0}^{k_m - N_{\text{gi}} - 1} \left(\left(\sum_{n_1=-N/2+1}^{N/2} D(n_1, l-1) e^{jn_1\omega_0(k-k_m+N_{\text{gi}})} \right) e^{-jn\omega_0 k} \right)$$

$$- \frac{h_m}{N} \sum_{k=0}^{k_m - N_{\text{gi}} - 1} \left(\left(\sum_{n_1=-N/2+1}^{N/2} D(n_1, l) e^{jn_1\omega_0(k-k_m)} \right) e^{-jn\omega_0 k} \right)$$

$$+ \frac{h_m}{N} \sum_{k=0}^{N-1} \left(\left(\sum_{n_1=-N/2+1}^{N/2} D(n_1, l) e^{jn_1\omega_0(k-k_m)} \right) e^{-jn\omega_0 k} \right). \tag{6}$$

Following the property of orthogonal basis function of $e^{-jn\omega_0 k}$, the last term in (6) can be written as

$$h_m e^{-jn\omega_0 k_m} D(n, l). \tag{7}$$

Since the term in (7) is only the frequency component at the nth carrier, clearly it still holds the carrier orthogonality. Nevertheless, the first and the second terms in (6), which are the summation within the interval $0 \le k \le k_m - N_{\text{gi}} - 1$ of an incomplete FFT window, yield leakage error whose frequency components contaminate all the carriers.

Now consider frequency components of all the multipaths. The orthogonal term at nth carrier becomes to

$$\sum_{m=0}^{M} h_m e^{-jn\omega_0 k_m} D(n, l) = H(e^{jn\omega_0}) D(n, l). \tag{8}$$

On the other hand, the first term in (6) for $k_m > N_{\text{gi}}$ yields ISI, which is the interference from the $(l-1)$th symbol period to the lth period. Let the representation of ISI be denoted as $E_{\text{s}}(n, l)$ in the frequency domain, then it can be expressed by

$$E_{\text{s}}(n, l) = \sum_{m=m_1}^{M} \frac{h_m}{N} \left(\sum_{k=0}^{k_m - N_{\text{gi}} - 1} \left(\sum_{n_1=-N/2+1}^{N/2} D(n_1, l-1) e^{jn_1\omega_0(k-k_m+N_{\text{gi}})} \right) e^{-jn\omega_0 k} \right), \tag{9}$$

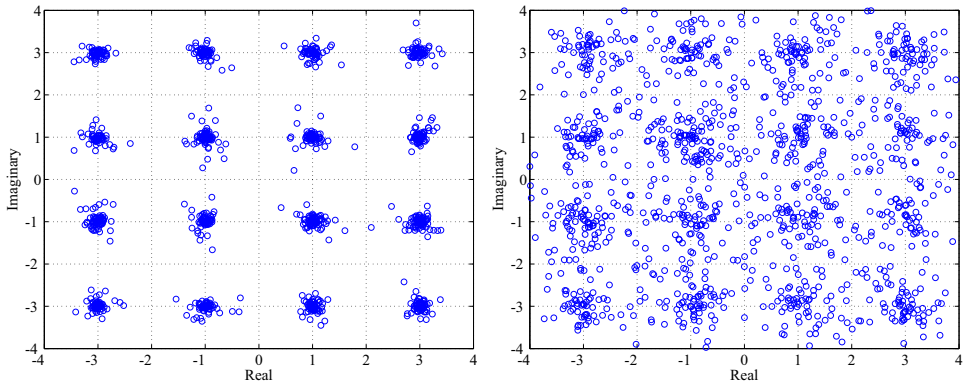
where m_1 is the smallest integer such that $k_{m_1} > N_{\text{gi}}$. Moreover, the effect of the second term in (6) leads to $E_{\text{c}}(n, l)$, which is the ICI term given by

$$E_{\text{c}}(n, l) = -\sum_{m=m_1}^{M} \frac{h_m}{N} \left(\sum_{k=0}^{k_m - N_{\text{gi}} - 1} \left(\sum_{n_1=-N/2+1}^{N/2} D(n_1, l) e^{jn_1\omega_0(k-k_m)} \right) e^{-jn\omega_0 k} \right). \tag{10}$$

Let the sum of ISI and ICI be denoted as a leakage error $E(n,l)$. Therefore, the frequency domain expression of the received signal in the lth symbol period is given by

$$Y(n,l) = H(e^{jn\omega_0})D(n,l) + \underbrace{E_{\mathrm{c}}(n,l) + E_{\mathrm{s}}(n,l)}_{E(n,l)} + V(n,l), \tag{11}$$

where $Y(n,l)$ and $V(n,l)$ are the frequency components of the received signal and noise at the nth carrier. It is clear that the leakage error $E(n,l)$ deteriorates the orthogonality of OFDM carriers and will cause large equalization error. A 16 QAM example with high signal to noise ratio (SNR=30dB) is shown in Fig.3. Besides the direct wave, there are two multipath interference waves. Fig.3(a) indicates the result of MMSE equalization where the delay taps are within GI. It is seen that its equalization error is very low, and can be removed by conventional error correction techniques. However, if one of the multipath interference has delay tap exceeding GI, the equalization error significantly increases even under high SNR situations. For example, just one interference with delay tap 1.25 times longer than GI increases the bit error rate (BER) up to 15% in Fig3(b). Therefore, it is important to reduce the influence of $E(n,l)$ to guarantee high communication performance.



(a) Equalization result when delay taps are within GI

(b) Equalization result when one of the delay taps exceeds GI.

Fig. 3. Examples of equalization with multipath interferences

3.1.2 Expressions of ICI and ISI

It is seen that only the coefficients h_m for $k_m > N_{\mathrm{gi}}$ remain in (9) and (10). Correspondingly, the sub-model of the multipaths exceeding GI can be expressed by z transform as

$$\Gamma(z) = h_{m_1} z^{N_{\mathrm{gi}}+1-k_{m_1}} + h_{m_1+1} z^{N_{\mathrm{gi}}+1-k_{m_1+1}} + \cdots + h_M z^{N_{\mathrm{gi}}+1-k_M}, \tag{12}$$

where $h_{m_1}, h_{m_1+1}, \cdots$ are the coefficients of the multipaths exceeding GI. Then performing inverse Fourier transform of $E_{\mathrm{s}}(n,l)$ in (9) and $E_{\mathrm{c}}(n,l)$ in (10) yields the signals of ISI and ICI in the time domain as

$$\varepsilon_{\mathrm{s}}(k,l) = \Gamma(z) d_{\mathrm{s}}(k,l), \tag{13}$$

$$\varepsilon_{\mathrm{c}}(k,l) = \Gamma(z) d_{\mathrm{c}}(k,l) \tag{14}$$

for $k = 0, 1, \cdots, N-1$, where $d_s(k,l)$ and $d_c(k,l)$ are the corresponding transmitted signals included in (9) and (10). They can be given by

$$d_s(k,l) = \begin{cases} d(k-1+(l-1)N_{tx}), & \text{for } -N+N_{gi}+1 \le k \le 0 \\ 0, & \text{for } k > 0 \end{cases}, \tag{15}$$

$$d_c(k,l) = \begin{cases} -d(k-N_{gi}-1+lN_{tx}), & \text{for } -N+N_{gi}+1 \le k \le 0 \\ 0, & \text{for } k > 0 \end{cases}. \tag{16}$$

On the other hand, $E_s(n,l)$ in (9) can be rewritten by

$$E_s(n,l) = \sum_{n_1=-N/2+1}^{N/2} H_s(n,n_1) D(n_1, l-1) \tag{17}$$

in the frequency domain, where $H_s(n,n_1)$ is

$$H_s(n,n_1) = \begin{cases} \sum_{m=m_1}^{M} \frac{h_m}{N} \frac{e^{-jn_1\omega_0(k_m-N_{gi})} - e^{-jn\omega_0(k_m-N_{gi})}}{1-e^{-j(n-n_1)\omega_0}}, & \text{for } n \ne n_1, \\ \sum_{m=m_1}^{M} \frac{k_m-N_{gi}}{N} h_m e^{-jn\omega_0(k_m-N_{gi})}, & \text{for } n = n_1. \end{cases}$$

Similarly, $E_c(n,l)$ in (10) is approximated by

$$E_c(n,l) = -\sum_{n_1=-N/2+1}^{N/2} H_c(n,n_1) D(n_1,l), \tag{18}$$

where $H_c(n,n_1)$ is given by

$$H_c(n,n_1) = \begin{cases} \sum_{m=m_1}^{M} \frac{h_m}{N} \frac{e^{-jn_1\omega_0 k_m} - e^{-jn\omega_0 k_m} e^{-j(n_1-n)\omega_0 N_{gi}}}{1-e^{-j(n-n_1)\omega_0}}, & \text{for } n \ne n_1, \\ \sum_{m=m_1}^{M} \frac{k_m-N_{gi}}{N} h_m e^{-jn\omega_0 k_m}, & \text{for } n = n_1. \end{cases}$$

From (17) and (18), the data in the frequency domain fulfil the following expression

$$\begin{aligned} \mathbf{HD}(l) &= \mathbf{Y}(l) - \mathbf{E}(l) - \mathbf{V}(l) \\ &= \mathbf{Y}(l) - \mathbf{E}_s(l) - \mathbf{H}_c\mathbf{D}(l) - \mathbf{V}(l), \end{aligned} \tag{19}$$

where $\mathbf{Y}(l)$, $\mathbf{D}(l)$, $\mathbf{E}(l)$, $\mathbf{E}_s(l)$ and $\mathbf{V}(l)$ are the vectors of FFT coefficients of the received signal, the information symbols, leakage error, ISI and noise in the lth period, respectively, and

$$\mathbf{H} = \begin{bmatrix} H\left(e^{j(-N/2+1)\omega_0}\right) & & \\ & \ddots & \\ 0 & & 0 \\ & & H\left(e^{jN/2\omega_0}\right) \end{bmatrix},$$

$$\mathbf{H}_c = \begin{bmatrix} H_c(-N/2+1, -N/2+1) & \cdots & H_c(-N/2+1, N/2) \\ \vdots & \ddots & \vdots \\ H_c(N/2, -N/2+1) & \cdots & H_c(N/2, N/2) \end{bmatrix}.$$

3.1.3 Statistical properties of ICI and ISI

Since $E_s(n,l)$ in (17) is only related to the information symbols in the $(l-1)$th symbol period, then from (2), $D(n,l)$ and $E_s(n,l)$ are uncorrelated, i.e.,

$$\lim_{L\to\infty}\frac{1}{L}\sum_{l=1}^{L}D^*(n,l)E_s(n,l) = \sum_{n_1=-N/2+1}^{N/2}\Big(\lim_{L\to\infty}\frac{1}{L}\sum_{l=1}^{L}D^*(n,l)D(n_1,l-1)\Big)H_s(n,n_1)=0. \tag{20}$$

Moreover, multiplying $E_c(n,l)$ in (18) by the conjugate information symbol $D^*(n,l)$ and using the results in (2) lead to the following result

$$\begin{aligned}\lim_{L\to\infty}\frac{1}{L}\sum_{l=1}^{L}D^*(n,l)E_c(n,l) &= -\sum_{n_1=-N/2+1}^{N/2}\Big(\lim_{L\to\infty}\frac{1}{L}\sum_{l=1}^{L}D^*(n,l)D(n_1,l)\Big)H_c(n,n_1)\\ &= -\bar{D}^2H_c(n,n) = -\bar{D}^2\sum_{m=m_1}^{M}\frac{k_m-N_{gi}}{N}h_m e^{-jnk_m\omega_0}.\end{aligned} \tag{21}$$

Following (21), it is clear that the longer the delay tap k_m, the greater the leakage error is. Therefore, the symbol equalization or interference compensation becomes more difficult.

Furthermore, from (11), the following equation

$$\begin{aligned}\frac{1}{L}\sum_{l=1}^{L}D^*(n,l)Y(n,l) &= H(e^{jn\omega_0})\frac{1}{L}\sum_{l=1}^{L}D^*(n,l)D(n,l)\\ &+\frac{1}{L}\sum_{l=1}^{L}D^*(n,l)\Big(E_s(n,l)+E_c(n,l)\Big)+\frac{1}{L}\sum_{l=1}^{L}D^*(n,l)V(n,l)\end{aligned} \tag{22}$$

holds true. Then by using the results of (2), (20) and (21), $\bar{H}(e^{jn\omega_0})$ defined in (23) can be obtained by (22) as follows.

$$\begin{aligned}\bar{H}(e^{jn\omega_0}) &= \lim_{L\to\infty}\frac{\frac{1}{L}\sum_{l=1}^{L}(D^*(n,l)Y(n,l))}{\frac{1}{L}\sum_{l=1}^{L}(D^*(n,l)D(n,l))}\\ &= H(e^{jn\omega_0})+\lim_{L\to\infty}\frac{\frac{1}{L}\sum_{l=1}^{L}(D^*(n,l)E_c(n,l))}{\frac{1}{L}\sum_{l=1}^{L}(D^*(n,l)D(n,l))}\\ &= \sum_{m=0}^{m_1-1}h_m e^{-jnk_m\omega_0}+\sum_{m=m_1}^{M}\left(1-\frac{k_m-N_{gi}}{N}\right)h_m e^{-jnk_m\omega_0}.\end{aligned} \tag{23}$$

Denote the IFFT coefficients of $\bar{H}(e^{jn\omega_0})$ as $\bar{h}_0, \bar{h}_1, \bar{h}_2, \cdots$, then the coefficients h_m can be obtained by

$$h_m = \begin{cases} \bar{h}_m, & \text{for } 0 \le k_m \le N_{\text{gi}}, \\ N\bar{h}_m/(N - N_{\text{gi}} + k_m), & \text{for } k_m > N_{\text{gi}}. \end{cases} \tag{24}$$

From (23) and (24), it can be seen that it is possible to estimate $H(e^{jn\omega_0})$ by using the properties of the leakage error even when the channel has long multipath interferences.

3.2 Channel identification algorithm

When several preamble or training symbols are available, (23) and (24) can give a batch channel identification only with computational complexity of $\mathcal{O}(N)$. When no successive training symbols are applicable for channel identification, the pilot symbols could be utilized in some conventional interpolation based channel estimation methods (Coleri et al., 2002; Nguyen et al., 2003). For example, provided that the scattered pilot symbols are assigned at P_nth carrier, then the symbol $D(P_n, l)$ at the pilot carrier P_n is known at the receiver, consequently, the estimation of $\bar{H}(e^{jn\omega_0})$

$$\hat{\bar{H}}(e^{jP_n\omega_0}) = \frac{\frac{1}{L}\sum_{l=1}^{L} D^*(P_n, l)Y(P_n, l)}{\frac{1}{L}\sum_{l=1}^{L} D^*(P_n, l)D(P_n, l)} \tag{25}$$

is obtained at the pilot carrier P_n. As for non-pilot carriers, if the pilot rate is high, a simple linear interpolation yields that

$$\hat{\bar{H}}(e^{jn\omega_0}) = \hat{\bar{H}}(e^{jP_{n,1}\omega_0}) + \frac{n - P_{n,1}}{P_{n,2} - P_{n,1}} \left(\hat{\bar{H}}(e^{jP_{n,2}\omega_0}) - \hat{\bar{H}}(e^{jP_{n,1}\omega_0}) \right), \tag{26}$$

where $P_{n,1}$ and $P_{n,2}$ are the number of two adjacent pilot carriers, $P_{n,1} \le n \le P_{n,2}$. Compared with the linear interpolation, the second order or high order interpolation methods could lead to more smooth interpolation. Furthermore, $\hat{H}(e^{jn\omega_0})$ can be determined by $\hat{\bar{H}}(e^{jn\omega_0})$.

Nevertheless, as mentioned in Section 3.1.1, the components of ICI and ISI contaminate all the carriers, and the frequency response function $H(e^{jn\omega_0})$ varies remarkably when the channel has long multipath interferences, as shown in Fig.4. As a result, neither the interpolation method nor equalization using the frequency selective diversity can yield satisfactory result if the pilot rate is not high enough.

We will consider some new information estimation and channel identification algorithm by making use of multiple receiver antennas and spectral periodograms whose ISI and ICI are compensated by the replica of leakage error.

3.2.1 Diversity of multiple antennas

Commonly, except the symbols at pilot carriers, the information symbols have to be estimated from the received signals for channel identification. Nevertheless, many existing symbol estimation methods cannot work well under the long multipath situations. It is seen that at

the nth carrier where $\left|H(e^{jn\omega_0})\right|$ is small, the orthogonal component in (6) attenuates to such a small value that symbol equalization becomes fragile to the noise and leakage error, even the error correction techniques might fail to correct the equalization errors at the carriers with small magnitude of frequency response. In order to overcome these difficulties, the diversity of multiple receiver antennas is used in the proposed algorithm. Let the total number of antenna elements be Q, correspondingly, the received signal at the qth antenna be denoted as $y_q(k)$, where $1 \leq q \leq Q$. Correspondingly, the frequency response function from the transmitter to the qth antenna is $H_q(e^{jn\omega_0})$, and its sub-model for the exceeding GI part is $\Gamma_q(z)$.

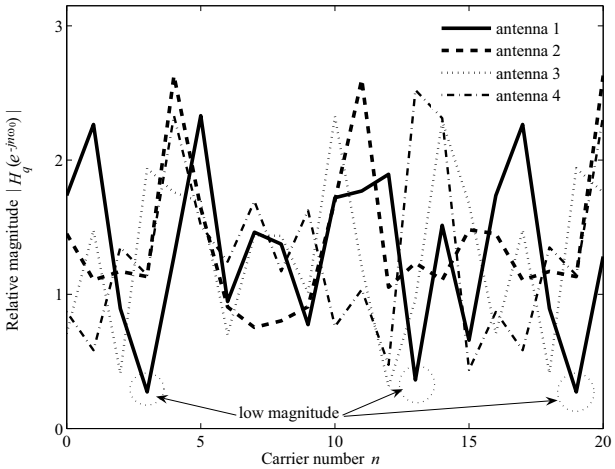


Fig. 4. Example of $|H_q(e^{-jn\omega_0})|$, $q = 1, \cdots, 4$

The relative magnitude of $H_q(e^{jn\omega_0})$ for $1 \leq q \leq 4$ and $0 \leq n \leq 20$ is illustrated in Fig.4. Consider $H_1(e^{-jn\omega_0})$ of the first antenna, notice that at the carriers $n = 3, 13, 19$ marked by circle the low $\left|H_1(e^{-jn\omega_0})\right|$ implies that the symbols are difficult to be estimated by frequency selective diversity of single antenna, since low magnitude of $H_1(e^{jn\omega_0})$ leads to a weak orthogonal component in the received signal $y_1(k)$. On the other hand, $H_2(e^{jn\omega_0})$ has larger magnitude at the carriers marked with circle, and can help the estimation of information symbols.

An approach had been discussed in (Sun et al., 2009) to perform symbol estimation per information carrier by selecting the largest magnitude $\left|H_q(e^{-jn\omega_0})\right|$ from the Q receiver antennas, where it had to perform ICI reduction and symbol estimation iteratively. Next, the more effective estimation approaches without iterative computation will be considered.

3.2.2 Estimation of information symbols

In the lth symbol period, let the frequency component of $y_q(k)$ at the nth carrier be denoted by $Y_q(n, l)$. It is calculated from $y_q(k)$ easily by using FFT algorithm within the FFT window $lN_{\text{tx}} \leq k < lN_{\text{tx}} + N$. From (19), the relation between the symbol vector and received signals

can be expressed by

$$\left(\mathbf{H}_q + \mathbf{H}_{\mathrm{c},q}\right)\mathbf{D}(l) = \mathbf{Y}_q(l) - \mathbf{E}_{\mathrm{s},q}(l) - \mathbf{V}_q(l), \tag{27}$$

where the subscript q indicates the antenna number. Two algorithms are developed to estimate the symbol vector $\mathbf{D}(l)$ in the lth period from the received signals and the estimated channel models.

In the $(l-1)$th symbol period, denote the estimates of information symbol as $\hat{D}(n, l-l)$, the frequency response as $\hat{H}_q(e^{jn\omega_0}, l-1)$, and its sub-model for the part exceeding GI as $\hat{\Gamma}_q(z, l-1)$, respectively. These estimates are used in the estimation of $\hat{D}(n,l)$ and $\hat{H}_q(e^{jn\omega_0}, l)$, and the affection of ISI is mitigated by estimating $\mathbf{E}_{\mathrm{s},q}(l)$ from $\hat{D}(n, l-l)$ and $\hat{\Gamma}_q(z, l-1)$ in (13). The received signal with ISI compensation in the frequency domain is indicated as $\bar{\mathbf{Y}}_q(l) = \mathbf{Y}_q(l) - \mathbf{E}_{\mathrm{s},q}(l)$.

A. Key magnitude selection based approach

Let $q_{\max}(n)$ be the antenna number such that

$$q_{\max}(n) = \arg\max_{q} \left(\left| \hat{H}_q(e^{jn\omega_0}, l-1) + \hat{H}_{\mathrm{c},q}(n,n) \right| \right). \tag{28}$$

It is seen that $|\hat{H}_{q_{\max}(n)}(e^{jn\omega_0}, l-1) + \hat{H}_{\mathrm{c},q_{\max}(n)}(n,n)|$ takes the maximum for $1 \le q_{\max}(n) \le Q$ at the nth carrier so that the strongest orthogonal component is used to estimate the information symbol $D(n,l)$. It means that the influence of ICI and ISI will be decreased through the technique of key magnitude selection (KMS). Define a matrix $\mathbf{H}_{\mathrm{kms}}(l)$ such that the nth diagonal entry is $\hat{H}_{q_{\max}(n)}(e^{jn\omega_0}, l-1) + \hat{H}_{\mathrm{c},q_{\max}(n)}(n,n)$, while the other entries in the nth row are the values of $\hat{H}_{\mathrm{c},q_{\max}(n)}(n, n_1)$. Furthermore, separate $\mathbf{H}_{\mathrm{kms}}(l)$ into the diagonal part $\mathbf{H}_{\mathrm{kms,dg}}(l)$ and the $\mathbf{H}_{\mathrm{kms,nodg}}(l)$. Then the estimate of $\mathbf{D}(l)$ can be given by

$$\hat{\mathbf{D}}(l) = \mathbf{H}_{\mathrm{kms}}^{-1}(l)\bar{\mathbf{Y}}_{\mathrm{kms}}(l) = \left(\mathbf{H}_{\mathrm{kms,dg}}(l) + \mathbf{H}_{\mathrm{kms,nodg}}(l)\right)^{-1}\bar{\mathbf{Y}}_{\mathrm{kms}}(l), \tag{29}$$

where $\bar{\mathbf{Y}}_{\mathrm{kms}}(l)$ is the corresponding vector of $Y_{q_{\max}(n)}(n,l)$. However, the computation of direct inverse $\mathbf{H}_{\mathrm{kms}}^{-1}(l)$ is time-consuming. In the proposed algorithm an approximation of the matrix inverse is considered.

Notice that the magnitude of the diagonal entries in $\mathbf{H}_{\mathrm{kms,dg}}(l)$ is larger than that of $\mathbf{H}_{\mathrm{kms,nodg}}(l)$ in the key magnitude selection, then the inverse $\mathbf{H}_{\mathrm{kms}}^{-1}(l)$ can be approximated by

$$\begin{aligned}
&\mathbf{H}_{\mathrm{kms}}^{-1}(l) = \left(\mathbf{H}_{\mathrm{kms,dg}}(l) + \mathbf{H}_{\mathrm{kms,nodg}}(l)\right)^{-1} \\
&= \left(\mathbf{H}_{\mathrm{kms,dg}}(l)\left(\mathbf{I} + \mathbf{H}_{\mathrm{kms,dg}}^{-1}(l)\mathbf{H}_{\mathrm{kms,nodg}}(l)\right)\right)^{-1} \\
&\approx \left(\mathbf{I} - \left(\mathbf{I} - \mathbf{H}_{\mathrm{kms,dg}}^{-1}(l)\mathbf{H}_{\mathrm{kms,nodg}}(l)\right)\mathbf{H}_{\mathrm{kms,dg}}^{-1}(l)\mathbf{H}_{\mathrm{kms,nodg}}(l)\right)\mathbf{H}_{\mathrm{kms,dg}}^{-1}(l).
\end{aligned} \tag{30}$$

where $\mathbf{H}_{\mathrm{kms,dg}}^{-1}(l)$ is just the reciprocal of the diagonal matrix $\mathbf{H}_{\mathrm{kms,dg}}(l)$. Therefore, the estimate of $\mathbf{D}(l)$ is obtained by multiplication of matrices and vectors, and the estimate of

information symbol $D(n,l)$ at nth carrier can be determined by hard decision, or other error correction techniques (Glover & Grant, 1998).

B. Maximal ratio combination based approach

Define a matrix $\mathbf{H}_{\text{mrc}}(l)$ and a vector $\bar{\mathbf{Y}}_{\text{mrc}}(l)$ whose entries of the nth row vector are the addition as follows

$$\sum_{q=1}^{Q}\left(H_q(e^{-jn\omega_0})+H_{\text{c},q}(n,n)\right)^*\left(\mathbf{H}_q(n)+\mathbf{H}_{\text{c},q}(n)\right), \tag{31}$$

$$\sum_{q=1}^{Q}\left(H_q(e^{-jn\omega_0})+H_{\text{c},q}(n,n)\right)^*\bar{Y}_q(n,l), \tag{32}$$

where $\mathbf{H}_q(n)$ and $\mathbf{H}_{\text{c},q}(n)$ are the nth row vector of $\mathbf{H}_q$ and $\mathbf{H}_{\text{c},q}$, respectively. If the phase of $H_q(e^{-jn\omega_0})+H_{\text{c},q}(n,n)$ is close to the true one, then the diagonal part $\mathbf{H}_{\text{mrc,dg}}(l)$ of $\mathbf{H}_{\text{mrc}}(l)$ will yield the dominate component of $\bar{\mathbf{Y}}_{\text{mrc}}(l)$ and lead to an effect of maximal ratio combination (MRC) (Burke et al., 2005). Therefore, similarly as (29), $\mathbf{D}(l)$ can be estimated by

$$\hat{\mathbf{D}}(l)=\mathbf{H}_{\text{mrc}}^{-1}(l)\bar{\mathbf{Y}}_{\text{mrc}}(l)=\left(\mathbf{H}_{\text{mrc,dg}}(l)+\mathbf{H}_{\text{mrc,nodg}}(l)\right)^{-1}\bar{\mathbf{Y}}_{\text{mrc}}(l), \tag{33}$$

where the inverse of $\mathbf{H}_{\text{mrc}}(l)$ is calculated by a similar approximation as in (30).

Compared with the KMS based approach, the MRC based approach uses all of the received signals' information to reduce the influence of the additive noise, whereas its performance depends on the phase accuracy of $\hat{H}_q(e^{-jn\omega_0})+\hat{H}_{\text{c},q}(n,n)$. A feasible choice is in the first several symbol periods to employ KMS based approach, which does not depend on the phase information so much, then to switch to the MRC based approach after the estimation error decreases to a low level.

3.2.3 Estimation of leakage error

The time domain sequence $\hat{d}(k)$ is calculated through IFFT of $\hat{D}(n,l)$, then $\hat{d}_{\text{c}}(k,l)$ can be obtained by (16). Using the estimation of $\hat{\Gamma}_q(z,l-1)$ and $\hat{d}_{\text{c}}(k,l)$, the values of $\varepsilon_{\text{c},q}(k,l)$ can be estimated by (14) in the time domain first. Consequently, $\hat{E}_{\text{c},q}(n,l)$ can be calculated from FFT of $\varepsilon_{\text{c},q}(k,l)$, and the leakage error $\hat{E}_q(n,l)$ can also be obtained by $\hat{E}_{\text{s},q}(n,l)+\hat{E}_{\text{c},q}(n,l)$.

3.2.4 Estimation of frequency response function

The channel model is estimated from the frequency component $Y_q(n,l)$, the symbol estimate $\hat{D}(n,l)$ and the replica of leakage error $\hat{E}(n,l)$. Consequently, it is important to remove the influence caused by the noise term in $Y_q(n,l)$, the estimation errors of $\hat{D}(n,l)$ and $\hat{E}(n,l)$. The phases of noise and estimation errors are usually random, then their influence can be mitigated through the smoothing effect of spectral periodograms (Pintelon & Schoukens, 2001). In the iteration of lth symbol period, the spectral periodograms $S_{DD}(e^{jn\omega_0},l)$ and $S_{DY}(e^{jn\omega_0},l)$ are defined as follows,

$$S_{DD}(e^{jn\omega_0},l)=\lambda_l S_{DD}(e^{jn\omega_0},l-1)+\hat{D}^*(n,l)\hat{D}(n,l), \tag{34}$$

$$S_{DY,q}(e^{jn\omega_0},l)=\lambda_l S_{DY,q}(e^{jn\omega_0},l-1)+\hat{D}^*(n,l)\left(Y_q(n,l)-\hat{E}_q(n,l)\right) \tag{35}$$

where λ_l is a forgetting factor over the range of $0 < \lambda_l < 1$. It is seen that the effects of noise and estimation errors are reduced when l becomes large. Using the estimates $S_{DD}(e^{jn\omega_0}, l-1)$ and $S_{DY,q}(e^{jn\omega_0}, l-1)$ in the iteration of $(l-1)$th symbol period, as well as the estimates $\hat{D}(n,l)$, $Y_q(n,l)$ and $\hat{E}_q(n,l)$ in the lth iteration, the estimates of (34) and (35) are obtained. Thus the estimate of frequency response function $\hat{H}_q(e^{jn\omega_0}, l)$ can be given by

$$\hat{H}_q(e^{jn\omega_0}, l) = \frac{S_{DY,q}(e^{jn\omega_0}, l)}{S_{DD}(e^{jn\omega_0}, l)}. \tag{36}$$

Moreover, $\hat{\Gamma}_q(z,l)$ can be updated by (12) using IFFT of $\hat{H}_q(e^{jn\omega_0}, l)$. Then let $l = l+1$ for the next iteration.

As mentioned previously, the frequency response function varies remarkably when the interferences have long delay taps. As a result, the side lobes often occur in the impulse response $\hat{h}_m$ of channel model due to the noise, the estimation errors of information symbols and leakage error, etc. Through setting a threshold between main lobe and side lobes, the effect of side lobes can be reduced to improve the convergence performance of channel identification and BER performance of symbol estimation (Hamazumi & Imamura, 2000).

3.2.5 Procedure of channel identification

In the identification algorithm, the estimation of $\hat{D}(n,l)$ is calculated from the received signal compensated by $\hat{E}_s(n,l)$ first, next the leakage error $\hat{E}(n,l)$ is estimated, then the channel frequency response is estimated from the spectral periodograms of the transmitted and received signals. The procedure of proposed algorithm can be summarized as follows.

Step 1. Let the initial values of $\hat{D}(n,0)$, $S_{DY,q}(e^{jn\omega_0}, 0)$, $S_{DD}(e^{jn\omega_0}, 0)$ be 0. Choose the initial values of $\hat{H}_q(e^{jn\omega_0}, 0)$, $\hat{\Gamma}_q(z,0)$, and let the iteration number be $l = 1$.

Step 2. Calculate $Y_q(n,l)$ from the received signal $y_q(k)$ within the FFT window $lN_{tx} \le k < lN_{tx} + N$ through FFT.

Step 3. Calculate $d_s(k,l)$ by (15), and $E_{s,q}(n,l)$ by FFT of $\varepsilon_{s,q}(k,l)$ in (13).

Step 4. Estimate $\hat{\mathbf{D}}(l)$ by (29) or (33), and determine $\hat{D}(n,l)$ by hard decision or some error correction techniques, respectively.

Step 5. Calculate $d_c(k,l)$ in (16), furthermore estimate the replica of ICI through FFT of $\hat{\varepsilon}_{c,q}(k,l)$ in (14).

Step 6. Calculate $S_{DY,q}(e^{jn\omega_0}, l)$ and $S_{DD}(e^{jn\omega_0}, l)$ by (34) and (35), respectively.

Step 7. Estimate $\hat{H}_q(e^{jn\omega_0}, l)$ from (36) and update the sub-model $\hat{\Gamma}_q(z,l)$ by (12). Let $l = l+1$, then return to Step 2 to repeat the iterations.

3.3 Algorithm features

The features of the proposed algorithm are summarized as follows.

(1) Periodograms can smooth the power spectra $S_{DY,q}(e^{jn\omega_0}, l)$ and $S_{DD}(e^{jn\omega_0}, l)$ so that the periodograms based identification algorithm can reduce the estimation error of $\hat{H}_q(e^{jn\omega_0}, l)$, which is caused by the estimation errors of $\hat{D}(n,l)$ and $\hat{E}_q(n,l)$, or the noise term in $Y_q(n,l)$ (Pintelon & Schoukens, 2001).

(2) Unlike some other algorithms in the time domain whose performance depends on the total number of parameters to be estimated (Ljung, 1999), the proposed algorithm estimates the frequency response function per carrier from the spectral periodograms and shows good convergence performance even for long impulse response of channel.

(3) By virtue of the multiple antennas' diversity, the stronger orthogonal component is used in KMS or MRC to estimate the information symbols, and it improves the performance of channel identification and symbol estimation significantly compared with the single antenna case.

It is noticed that the purpose of using multiple antennas is just the utilization of the stronger orthogonal component at each carrier, the performance of the proposed algorithm does not depend on the total number of antenna elements as much as the conventional spatial equalizers based on antenna diversity (Higuchi & Sasaoka, 2004; Hori et al., 2003). Moreover, the switching between KMS and MRC improves the algorithm performance under the low pilot rate or low SNR environment, and the proposed algorithm can easily be combined with some error correction techniques to obtain better BER performance.

(4) The forgetting factor is used in periodogram estimation so that the algorithm can also deal with slow time-varying channels. A small forgetting factor has adaptability to quick channel variation and large BER, whereas a large one is used in low SNR environment for spectral smoothing.

(5) When the interference that exceeds GI is strong, the convergence of channel identification and BER performance can be improved by appropriately choosing the side lobe threshold to reduce the influence of side lobes.

(6) The main computation only requires FFT, matrix multiplication, division of periodograms, therefore, the algorithm has less computational complexity and can easily be utilized in the practical applications.

In each iteration, besides the computation of FFT, the proposed algorithm needs the following calculations: $2(k_M - N_{\text{gi}})^2 q$ multiplications to estimate ICI and ISI, about N^2 multiplications and divisions for updating $\mathbf{H}_{\text{kms}}(l)$ or $\mathbf{H}_{\text{mrc}}(l)$, about $2N^2$ multiplications to estimate the information symbols, $(q+1)N$ multiplications to estimate periodograms, qN divisions to estimate the frequency response functions. It is seen that the main computation concentrates on the estimation of information symbols, whereas the channel identification is very simple in the frequency domain. By contrast with the computational complexity of RLS algorithm, besides the estimation of information symbols in RLS algorithm, the recursive identification requires about $\mathcal{O}(k_M^2 N_{\text{tx}})$ multiplications for one symbol period. It is clear that the proposed identification algorithm has less complexity than RLS, especially for large k_M. Though LMS only requires $\mathcal{O}(2k_M N_{\text{tx}})$ multiplications for channel identification, its convergence rate is much slower than RLS (Balakrishnan et al., 2003).

3.4 Numerical simulation examples

3GPP 2.5MHz OFDM transmission with 16QAM modulation is used in the examples where the FFT size $N = 256$, GI length $N_{\text{gi}} = N/4 = 64$ (3GPP, 2006). Moreover, the number of carriers is 256, the total number of receiver antennas is $Q = 2$, and their distance is $1/2$ of the wave length. The noise is assumed as an additive Gaussian white noise.

3.4.1 Example of a simple channel model

Besides the direct wave, there are three multipath interferences in the transmission channel (Higuchi & Sasaoka, 2004; Hori et al., 2003). The coefficients of the waves are illustrated in Table.1. Let the SNR= 20dB. The simulation is performed under 6 conditions of pilot rates:

Wave	Power (dB)	Phase	DOA	Delay tap
Direct	0	0	$\pi/8$	0
Interference 1	-3	$-\pi/6$	$-\pi/4$	$3N_{\rm gi}/4$
Interference 2	-5	$\pi/4$	$\pi/3$	$5N_{\rm gi}/4$
Interference 3	-5	$-\pi/3$	$-\pi/6$	$7N_{\rm gi}/4$

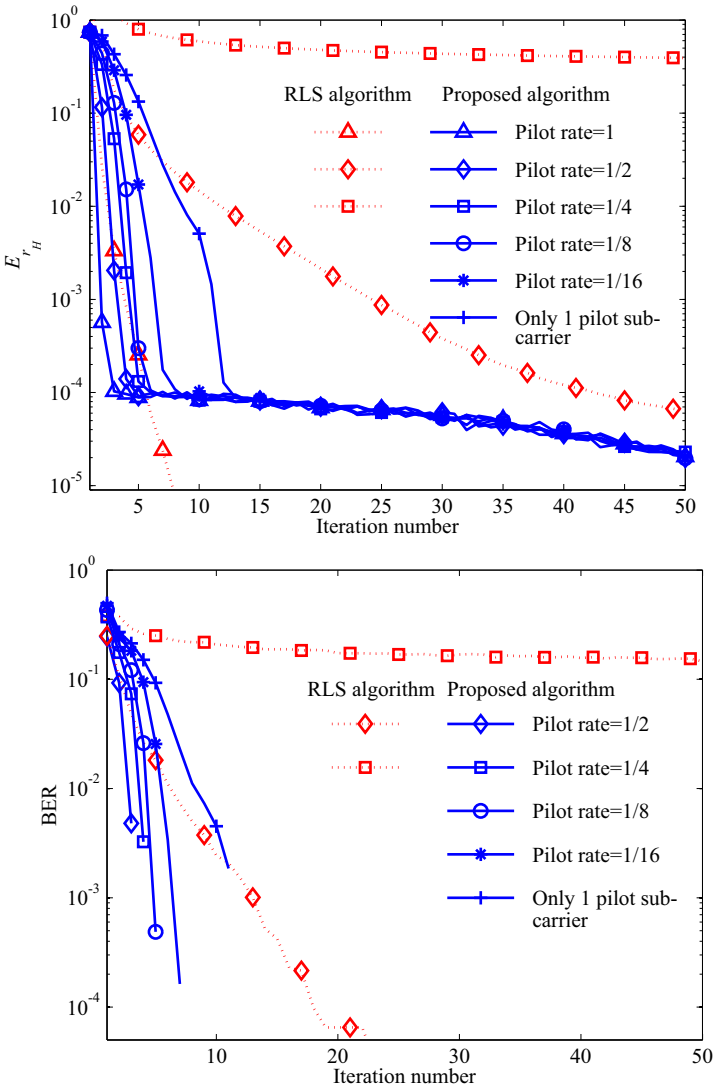
Table 1. Simulation conditions

The successive training symbols are available for identification; the pilot rates are 1/2, 1/4, 1/8, 1/16, respectively; and a severe case where only one pilot carrier is known to remove the ambiguity of channel identification and symbol estimation. At the pilot carrier, the value of $\hat{D}(n,l)$ is given by the corresponding true value of pilot symbol, while at the other carriers, the value of $D(n,l)$ has to be estimated from the received signals. Channel identification is started from the initial values of $\hat{H}(e^{jn\omega_0},0)=1, \hat{\Gamma}(z,0)=0$. For the comparison of estimation errors, the square error of channel identification Er_H defined by

$$Er_H = \frac{\sum_q \sum_{n=0}^{N-1} \left| \hat{H}_q(e^{jn\omega_0}) - H_q(e^{jn\omega_0}) \right|^2}{\sum_q \sum_{n=0}^{N-1} \left| H_q(e^{jn\omega_0}) \right|^2}, \tag{37}$$

is illustrated in Fig.5(a), and BER curves of the estimated symbols are illustrated in Fig.5(b), respectively. They show that the algorithm works well even for few pilot carrier cases. In Fig.5(b), BER of the estimated symbols decreases to 0 after several iterations, whereas it is larger than 0.5 at low pilot rate in the first iteration due to the initial value of channel identification is quite different from the true one. It implies that the algorithm converges even under the severe initial conditions. The BER curves are plotted in Fig.5(b) when the channel estimate is used for symbol estimation. It is seen that the good BER performance can be guaranteed if the influence of ISI and ICI caused by the long multipath interferences is compensated by the replica of leakage error.

Since RLS algorithm is often used for channel identification in the previous methods, the results of RLS algorithm are also shown in Fig.5(a) and Fig.5(b) for comparison with the proposed algorithm. They are obtained under the same simulation conditions. In RLS algorithm, the recursion is performed per sampling instant to estimate the parameters of h_m in (3) by using the latest samples of $y(k)$ and $d(k)$, thus RLS updates the estimates $N_{\rm tx}=320$ times during 1 iteration in the proposed algorithm. In Fig.5(a), if several successive training periods are available, i.e., the true values of $d(k)$ can be used for channel identification directly, it is seen that RLS algorithm can yield a small error by using the true $d(k)$ while its computational load is heavier than that of the proposed algorithm. However, when the training symbols are unavailable, $\hat{d}(k)$ has to be estimated for channel identification, the error of $\hat{d}(k)$ deteriorates the identification performance of RLS algorithm. For example, when the pilot rate is 1/2, the convergence of channel identification and the symbol estimation



(b) BER of estimated symbols.

Fig. 5. Estimation result (average of 30 simulation runs)

used RLS estimation is much slower than that of the algorithm in frequency domain, and its performance becomes very poor when the pilot rate is 1/4, whereas the proposed algorithm works well since it uses spectral periodograms.

3.4.2 Channel identification versus noise

Let the pilot rate be 1/8, and SNR be changed from 10dB to 40dB, the other conditions be the same as those in Section 3.4.1. Er_H plotted in Fig.6 shows that the proposed algorithm converges even for low SNR conditions.

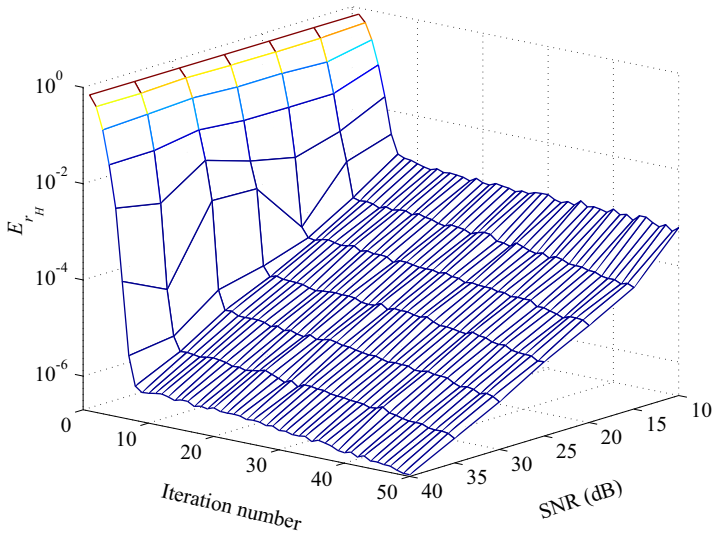


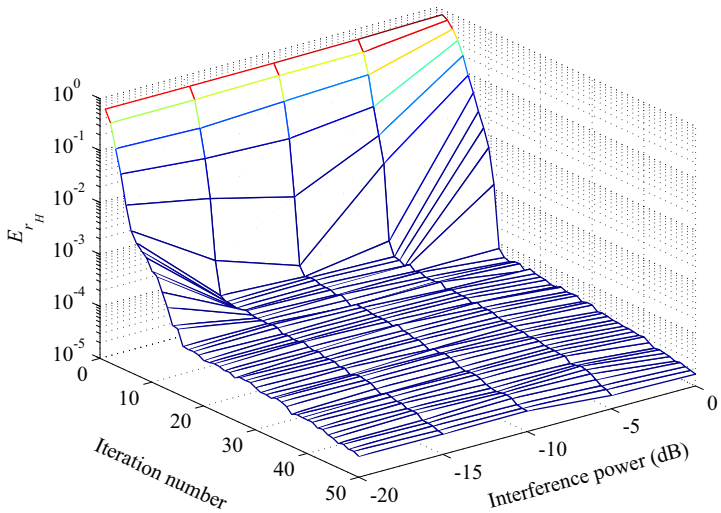
Fig. 6. Channel estimation error versus SNR

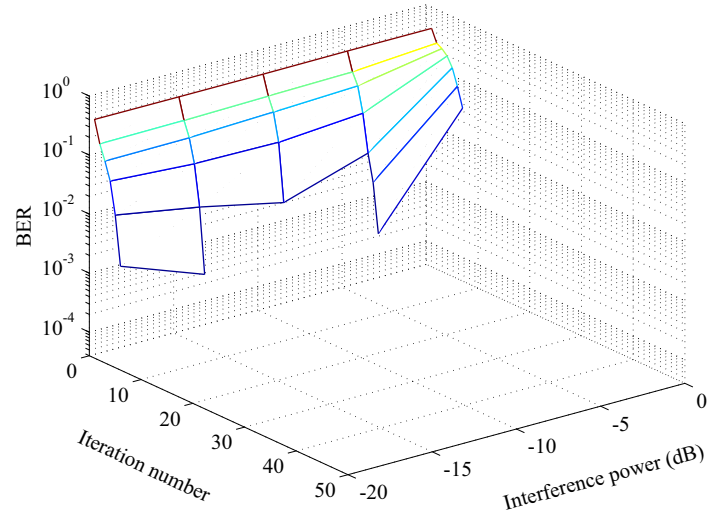
3.4.3 Channel identification versus interference power

Let the pilot rate be 1/8, and the power of interference 3 be changed from 0dB to -20dB, the other conditions be the same as those in Section 3.4.1. Er_H and BER versus interference power are illustrated in Fig.7(a) and Fig.7(b), respectively. It is seen that the interferences exceeding GI with high path gain cause severe ICI and ISI, while their information is not so fragile to the side lobes caused by noise, ICI and ISI. Therefore, the convergence of the channel estimation for the interference path with high gain is a little faster than the convergence for the interference with low path gain, as shown in Fig.7(a), and the convergence of channel estimation helps to compensate the influence of ICI and ISI. Though there are two strong multipath interferences exceeding GI, both Er_H and BER are decreased to a considerable low level after just about several iterations.

3.4.4 Channel identification versus total number of antennas

Let the pilot rate be 1/8. In order to investigate the influence of Q on the algorithm performance, the total number Q of antennas is chosen as $Q = 1$, 2, 3 and 4 respectively. The other simulation conditions are the same as those in Section 3.4.1. The curves of Er_H and





(b) Bit error rate

Fig. 7. Estimation error versus interference power

BER are plotted in Fig.8. If only a single antenna 1 is used for symbol estimation, as illustrated in Fig.4, although the estimation error of $H_1(e^{jn\omega_0})$ decreases to a low level after 10 iterations, the low magnitude of $H_1(e^{jn\omega_0})$ leads to weak orthogonal component in the received signal, as a result, the BER remains high for $Q = 1$. On the other hand, both the errors E_{r_H} and BER for $Q = 2$, 3 and 4 are very low since the strong orthogonal components can be used in symbol estimation, and just 2 antennas can yield good performance in this example. It is seen that though BER is large in the first several iterations, its influence is mitigated in periodograms so that channel identification can provide an effective channel model for equalization.

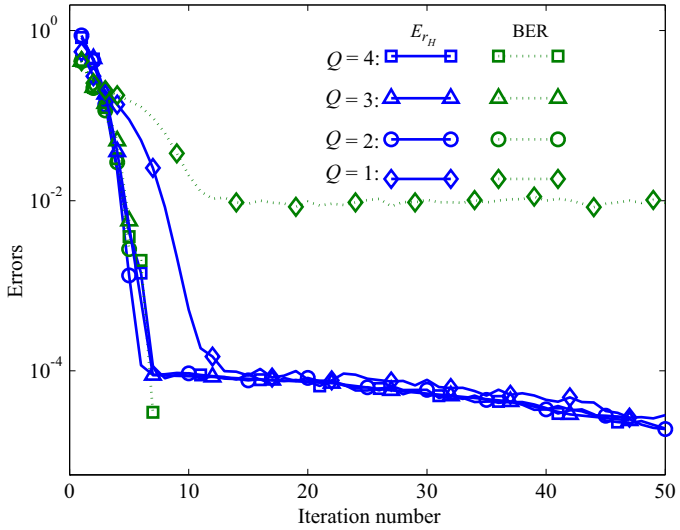


Fig. 8. Channel estimation error and BER versus antenna number Q

3.4.5 Channel identification versus FFT size N

The effect of FFT size N on the channel identification is considered in the simulation. Here N is chosen as 64, 128, 256 and 512, respectively, the GI length is $N_{\text{gi}} = N/4$. Besides the direct path, the delay taps of 3 interference paths are $3N_{\text{gi}}/4$, $5N_{\text{gi}}/4$, $7N_{\text{gi}}/4$, respectively. The coefficients of interference power, phase, DOA are given in Table 1, and the other simulation conditions are the same as those in Section 3.4.1. The channel estimation error E_{r_H} under the 4 cases of FFT size N is illustrated in Fig. 9.

Following the expressions in (9) and (10), or in (17) and (18), it is seen that the effects of ICI and ISI decrease a little with increasing FFT size N in the spectral periodogrames of $S_{DY,q}(e^{jn\omega_0}, l)$. Consequently, the error of channel estimation becomes lower for large N since the effects of side lobe caused by ICI and ISI reduce with increasing FFT size N, and the proposed algorithm yields good BER performance under the given simulation conditions.

3.4.6 Identification of time-varying channel

Let the pilot rate be 1/8, the power, phase and DOA of Interference 3 be changed at every 10 symbol periods so that the channel is time-varying. The power profile is shown in Fig.10(a).

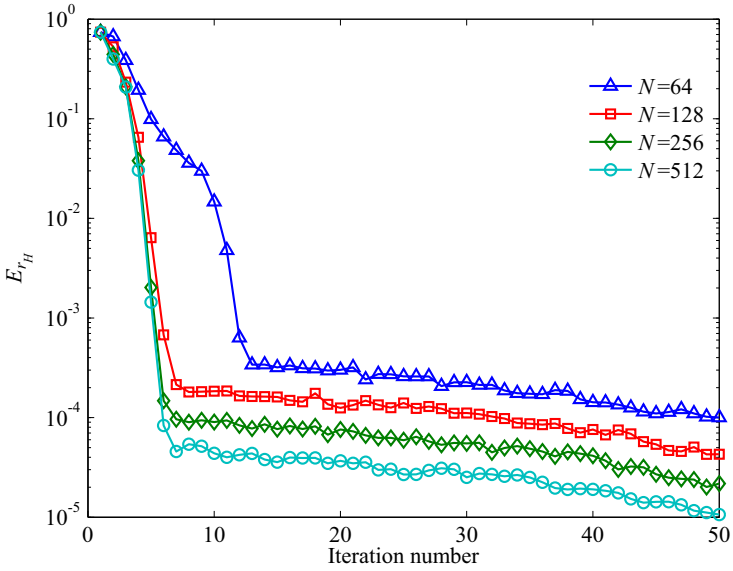


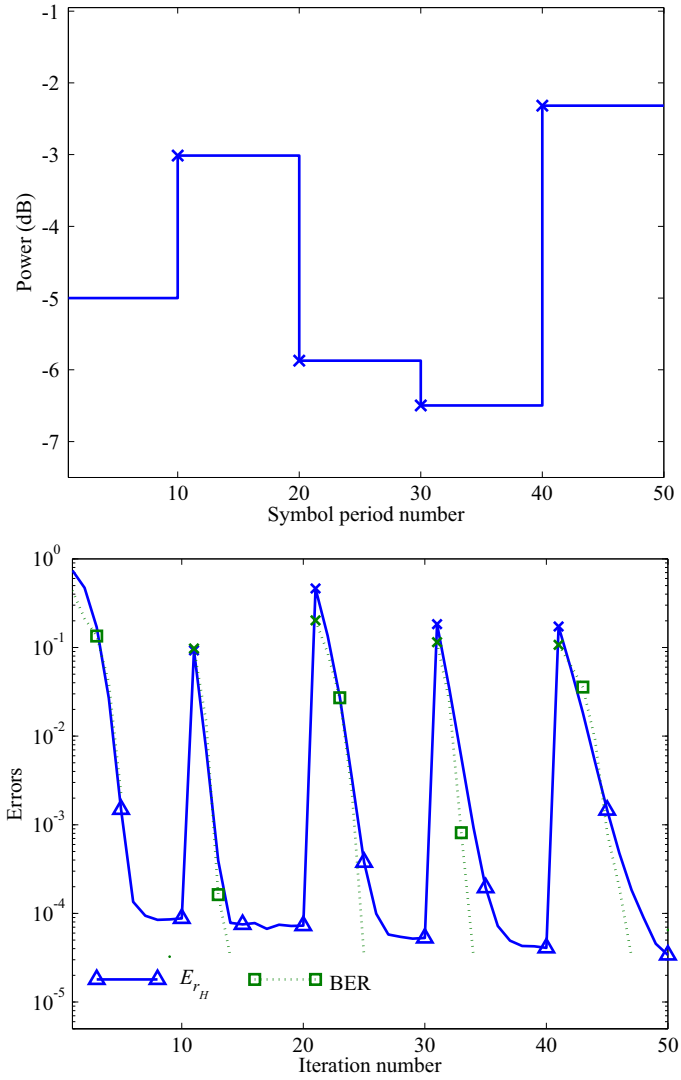
Fig. 9. Channel estimation error versus FFT size N

For the variation of channel, the forgetting factor $\lambda_l = \min\{0.075 \times 1.05^l, 0.75\}$ is used. At the first several iterations, small λ_l is selected to mitigate the influences of high estimated symbols' BER and the effect of side lobes. With decreasing of BER, λ_l is increased gradually to smooth the periodograms. The results of Er_H, BER are illustrated in Fig.10(b). Though the channel varies quickly, the prompt reduction of errors shows that the proposed algorithm can also work well for time-varying channels.

3.4.7 Channel identification of COST 207 model

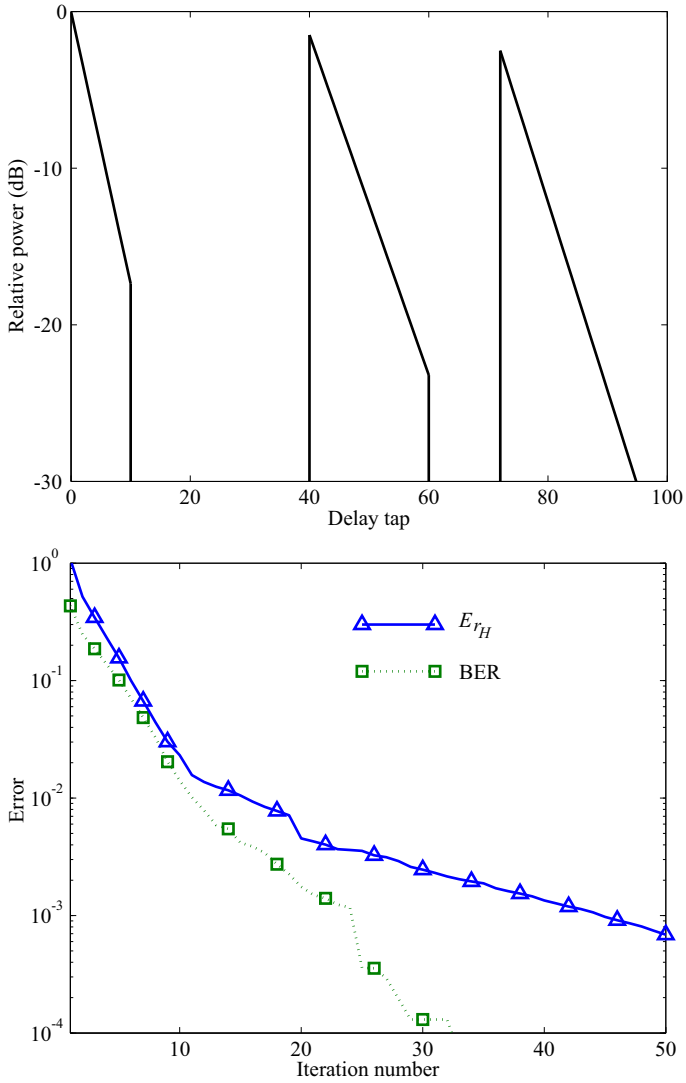
Let the pilot rate be $1/8$. Assume that the delay profile of the multipath in a hill area is a COST 207 model (European Communities, 1989). There are eleven waves with delay time $0 \le k_m \le 10$ and power $e^{-\frac{k_m}{2.5}}$, twenty one waves with delay time $40 \le k_m \le 60$ and power $0.7079e^{-\frac{k_m-40}{4}}$, fifteen waves exceeding GI with delay time $72 \le k_m \le 96$ and power $0.5623e^{-\frac{k_m-72}{3.6}}$, and $m = 0$ denotes the direct wave. The total power of interferences exceeding GI is -1.29dB. The DOA of multipath waves are generated randomly, and the other conditions are the same as in Section 3.4.1.

As illustrated in Fig.11(a), the received signal suffers from strong multipath interferences, as a result, BER is about 0.4 without interference compensation. In the simulation, the side lobe threshold is chosen as $\max\{0.1 \times 0.98^l, 0.005\}$ to reduce the influence of side lobe. At the first several iterations, a large side lobe threshold is selected, whereas the side lobe threshold is decreased gradually to deal with weak multipath interferences. The corresponding Er_H and BER of estimated symbols are shown in Fig.11(b). Though the channel has strong multipath interferences, Er_H decreases from 1.0 to 0.0025, BER decreases from 0.4 to 0 in about 30 iterations.



(b) Channel estimation error and BER

Fig. 10. Estimation errors for time-varying channel



(b) Channel estimation error and BER of estimated symbols

Fig. 11. Estimation errors (average of 30 simulation runs).

4. Identification of channel with limited bandwidth

In some large capacity OFDM systems such as digital terrestrial television broadcasting, the carriers far away from the frequency of central band do not convey any information data in order to simplify the design of filter with sharp cut-off performance and not to interference the adjacent communication channels, thus the transmitted signal is restricted within a specified frequency band. Due to the dynamic modes of the transmission channel cannot be excited beyond the signal band, channel identification becomes a very difficult problem (Ljung, 1999).

On the other hand, if the adaptive algorithms for OFDM system has feedback element, not only the channel information inside the signal band, but also the outside band is important to the processing performance of system stability and convergence rate. When the delay taps are very short and the width of outside band is much narrower than the signal band, extrapolation may exploit a little information outside the signal band from the information inside the band (Hamazumi et al., 2000). This problem had been discussed in the time domain (Sun & Sano, 2005; Ysebaert et al., 2004), however, it is suffered from considerable computational complexity due to the nonlinear optimization or operation of high dimension data matrices. So a frequency domain approach is investigated to decrease the computational complexity (Sun & Sano, 2007).

4.1 Signal bandwidth

In the lth transmission symbol period, the symbol $D(n,l)$ of OFDM signal with limited bandwidth is as follows.

$$D(n,l) = \begin{cases} \text{Symbol data}(\neq 0), & \text{for } |n| \leq N_1 \\ 0, & \text{for } |n| > N_1 \end{cases} \tag{38}$$

where $N_1 < N/2$. It can be seen that the carriers whose distance from the central carrier are more than N_1 do not carry any information data, hence the spectrum of transmitted signal $d(k)$ in an effective symbol period, i.e. $lN_{\text{tx}} \leq k < lN_{\text{tx}} + N$, is limited to $|n| \leq N_1$, whereas the spectral density outside signal band, i.e. for $|n| > N_1$, becomes to 0. It implies that $d(k)$ and its corresponding received signal $y(k)$ for $lN_{\text{tx}} \leq k < lN_{\text{tx}} + N$ do hold little information about the channel dynamics outside the signal band.

4.2 Fourier analysis of specified signals

Two signals are constructed from the original transmitted and received signals as follows:

$$u(k) = d(k - N_{\text{gi}}) - d(k - N_{\text{gi}} - N), \tag{39}$$

$$x(k) = y(k - N_{\text{gi}}) - y(k - N_{\text{gi}} - N). \tag{40}$$

As an example, the signal $u(k)$ is illustrated in Fig.12, where K is an integer satisfying $N_{\text{gi}} < K + N_{\text{gi}} \leq N$, e.g., $K = N/2$. Moreover, the expressions of $u(k)$, $x(k)$ and their relation can also be constructed similarly as $d(k)$ and $y(k)$. From the feature of guard interval, $u(k) = 0$ holds for $lN_{\text{tx}} - N_{\text{gi}} \leq k < lN_{\text{tx}}$. Then substituting (1) into the expression of $u(k)$ yields that

$$u(k) = \sum_{n=-N_1}^{N_1} D(n,l) e^{jn\omega_0(k - lN_{\text{tx}} - N_{\text{gi}})} - \sum_{n=-N_1}^{N_1} D(n,l-1) e^{jn\omega_0(k - lN_{\text{tx}})}$$

$$= \sum_{n=-N_1}^{N_1} \left(D(n,l)e^{-jn\omega_0 N_{\text{gi}}} - D(n,l-1) \right) e^{jn\omega_0(k-lN_{\text{tx}})},$$

$$\text{for } lN_{\text{tx}} \le k < K + lN_{\text{tx}}. \tag{41}$$

Next consider the signal $x(k)$. Let its component corresponding to interference m be denoted

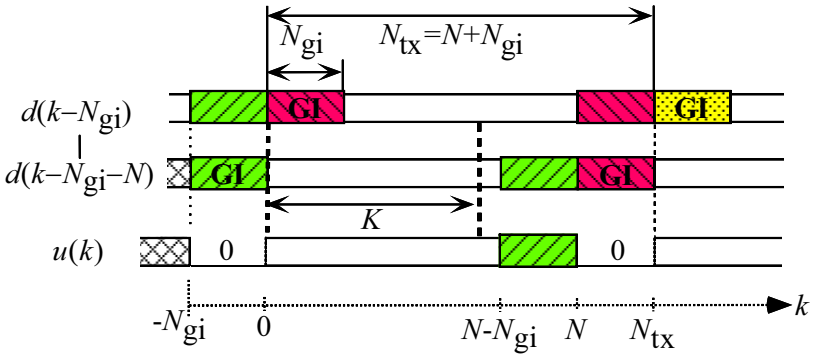


Fig. 12. Illustration of signal $u(k)$

by $x_m(k)$, then omitting the noise term for the simplicity of notation, $x_m(k)$ and $x(k)$ can be expressed by

$$x_m(k) = h_m u(k-k_m), \;\; x(k) = \sum_{m=0}^{M} x_m(k) \tag{42}$$

Moreover, in the lth symbol period, $x_m(k)$ becomes to

$$x_m(k) = \begin{cases} h_m \sum\limits_{n=-N_1}^{N_1} \left(D(n,l)e^{-jn\omega_0 N_{\text{gi}}} - D(n,l-1) \right) e^{jn\omega_0(k-lN_{\text{tx}}-k_m)}, \\ \qquad\qquad \text{for } k_m \le k - lN_{\text{tx}} < K \\ 0, \qquad\qquad \text{for } 0 \le k - lN_{\text{tx}} < k_m \end{cases} \tag{43}$$

On the other hand, let the Fourier transform of $u(k)$ in $lN_{\text{tx}} \le k < K + lN_{\text{tx}}$ be given by

$$U(n,l) = \sum_{k=lN_{\text{tx}}}^{K+lN_{\text{tx}}-1} u(k)e^{-jn\omega_0(k-lN_{\text{tx}})} \tag{44}$$

for $n = -N/2+1, \cdots, N/2$, and similarly the Fourier transform of $x(k)$, then the following frequency properties of signals $u(k)$ and $x(k)$ satisfy the following equation:

$$H(e^{jn\omega_0})U(n,l) = X(n,l) + \sum_{m=0}^{M} h_m \left(E_{m,1}(n,l) + E_{m,2}(n,l) \right) \tag{45}$$

where $E_{m,1}(n,l)$ and $E_{m,2}(n,l)$ are the leakage terms to the nth frequency point from the other components, the nth frequency component itself, respectively. Their theoretical

representations are expressed by

$$E_{m,1}(n,l)=e^{-jn\omega_0 k_m}\sum_{\substack{\bar{n}=-N_1\\ \bar{n}\neq n}}^{N_1}\Big(D(\bar{n},l)e^{-j\bar{n}\omega_0 N_{\mathrm{gi}}}-D(\bar{n},l-1)\Big)\cdot\frac{e^{-j(n-\bar{n})\omega_0(K-k_m)}-e^{-j(n-\bar{n})\omega_0 K}}{1-e^{-j(n-\bar{n})\omega_0}}, \tag{46}$$

$$E_{m,2}(n,l)=k_m e^{-j\omega_0 k_m}\Big(D(n,l)e^{-jn\omega_0 N_{\mathrm{gi}}}-D(n,l-1)\Big). \tag{47}$$

4.3 Spectra estimation

Let the power spectrum of $u(k)$ be estimated by

$$S_{UU}(n,l)=\frac{1}{l}\sum_{l_1=1}^{l}U^*(n,l_1)U(n,l_1), \tag{48}$$

and $S_{UE_{m,1}}(n,l)$ and $S_{UE_{m,2}}(n,l)$ are defined in the similar formula. Then $S_{UU}(n,l)$ can be approximated as

$$S_{UU}(n,l)\approx 2\overline{D}^2\sum_{\substack{\bar{n}=-N_1\\ \bar{n}\neq n}}^{N_1}\frac{1-\cos(n-\bar{n})\omega_0 K}{1-\cos(n-\bar{n})\omega_0}+2K^2\overline{D}^2$$

for $|n|\leq N_1$ when l is large enough, and

$$S_{UU}(n,l)\approx 2\overline{D}^2\sum_{\bar{n}=-N_1}^{N_1}\frac{1-\cos(n-\bar{n})\omega_0 K}{1-\cos(n-\bar{n})\omega_0} \tag{49}$$

for $N_1<|n|<N/2$. Meanwhile, $S_{UE_{m,1}}(n,l)$ satisfies

$$S_{UE_{m,1}}(n,l)\approx\overline{D}^2\sum_{\substack{\bar{n}=-N_1\\ \bar{n}\neq n}}^{N_1}\left(\frac{1-e^{-j(n-\bar{n})\omega_0 K}}{1-\cos(n-\bar{n})\omega_0}\left(e^{-jn\omega_0 k_m}-e^{-j\bar{n}\omega_0 k_m}\right)\right). \tag{50}$$

Furthermore, the spectral leakage error $S_{UE_{m,2}}(n,l)$ for $|n|\leq N_1$ is

$$S_{UE_{m,2}}(n,l)\approx 2e^{-jn\omega_0 k_m}\overline{D}^2 K k_m, \tag{51}$$

while for $N_1<|n|<N/2$ it turns to

$$S_{UE_{m,2}}(n,l)=0. \tag{52}$$

Following (45), the relation between spectra of $u(k)$, $x(k)$ and the frequency property $H(e^{jn\omega_0})$ is summarized in (53):

$$H(e^{jn\omega_0})S_{UU}(n,l)\approx S_{UX}(n,l)+\sum_{m=0}^{M}h_m\Big(S_{UE_{m,1}}(n,l)+S_{UE_{m,2}}(n,l)\Big). \tag{53}$$

Using the symbol estimation of $D(n,l)$, the signals $d(k)$ as well as $u(k)$ are estimated. Then, $S_{UU}(n,l)$ and $S_{UX}(n,l)$ in (53) can be estimated from $u(k)$ and $x(k)$ directly. On the other hand, from (50)-(52), the terms of spectral leakage error $S_{UE_{m,1}}(n,l)$ and $S_{UE_{m,2}}(n,l)$ can be calculated beforehand without using observation data and the information of channel dynamics. Consequently, it is possible to estimate the channel property outside the signal band from signals $u(k)$ and $x(k)$ if $S_{UX}(n,l)$ is compensated by $S_{UE_{m,1}}(n,l)$ and $S_{UE_{m,2}}(n,l)$.

On the other hand, when the channel is time-varying, a forgetting factor λ can be used to estimate $S_{UU}(n,l)$ and $S_{UX}(n,l)$

$$S_{UU}(n,l) = \lambda S_{UU}(n,l-1) + U^*(n,l)U(n,l), \tag{54}$$

$$S_{UX}(n,l) = \lambda S_{UX}(n,l-1) + U^*(n,l)X(n,l) \tag{55}$$

respectively, where $0 < \lambda < 1$.

4.4 Channel identification algorithm

Following (53), the estimation of channel model can be deduced as

$$\hat{H}(e^{jn\omega_0}) = \frac{S_{UX}(n,l)}{S_{UU}(n,l)} + \sum_{m=0}^{M} \hat{h}_m \frac{S_{UE_{m,1}}(n,l) + S_{UE_{m,2}}(n,l)}{S_{UU}(n,l)}. \tag{56}$$

Notice that the estimates $\hat{h}_m$, the coefficients of IFFT of $\hat{H}(e^{jn\omega_0})$, the estimation can be performed in the following iterative form as

$$\hat{H}^{(i+1)}(e^{jn\omega_0}) = \frac{S_{UX}(n,l)}{S_{UU}(n,l)} + \sum_{m=0}^{M} \hat{h}_m^{(i)} \frac{S_{UE_{m,1}}(n,l) + S_{UE_{m,2}}(n,l)}{S_{UU}(n,l)}, \tag{57}$$

where i is the iteration number, and $\hat{h}_m^{(0)}$ can be chosen as the coefficients of $S_{UX}(n,l)/S_{UU}(n,l)$. When the channel dynamics does not vary too fast, the recursive estimation can also be given by using the estimates $\hat{h}_m^{(l-1)}$, which are estimated in the last symbol period:

$$\hat{H}^{(l)}(e^{jn\omega_0}) = \frac{S_{UX}(n,l)}{S_{UU}(n,l)} + \sum_{m=0}^{M} \hat{h}_m^{(l-1)} \frac{S_{UE_{m,1}}(n,l) + S_{UE_{m,2}}(n,l)}{S_{UU}(n,l)}. \tag{58}$$

The main numerical computation in the identification algorithm is just FFT to estimate the signal $u(k)$, power spectra $S_{UU}(n,l)$, $S_{UX}(n,l)$, division of $\left(S_{UE_{m,1}}(n,l) + S_{UE_{m,2}}(n,l)\right)$ and $S_{UU}(n,l)$, while the two leakage error terms can be pre-calculated, division of $S_{UX}(n,l)$, and $S_{UU}(n,l)$, IFFT $\hat{H}(e^{jn\omega_0})$ to calculate $\hat{h}_m$. Furthermore, the computational complexity does not increase too much even though the interference delay taps get longer. So the identification algorithm can be easily implemented, and combined with other adaptive processing techniques.

On the other hand, $\hat{H}^{(}e^{jn\omega_0})$ inside the signal band can also be given by the spectra of $S_{DY}(n,l)$ and $S_{DD}(n,l)$

$$\hat{H}^{(}e^{jn\omega_0}) = \frac{S_{DY}(n,l)}{S_{DD}(n,l)}, \text{ for } \left|n\right| \leq N_1, \tag{59}$$

where $S_{DY}(n,l)$ and $S_{DD}(n,l)$ are calculated from the received signal $y(k)$ and the symbol estimates $\hat{D}(n,l)$ without using $\left(S_{UE_{m,1}}(n,l)+S_{UE_{m,2}}(n,l)\right)$. Furthermore, if the channel has not too long multipath interferences, interpolating channel information from the pilot carriers to their adjacent carriers is applicable to channel identification inside the signal band, and may reduce the influence caused by the estimation error of information symbols $D(n,l)$.

4.5 Numerical simulation examples

In the simulation, the OFDM information symbols $D(l,n)$ are 64QAM, the FFT/IFFT length is $N = 2048$, the guard interval is $N_{\text{gi}} = N/4$, and $N_1 = 600$. It means that the number of active carriers is 1201, and the signal band is only about 3/5 of the full band width, hence identification of such an OFDM channel is a very difficult problem. There are 6 symbol transmission periods per signal frame, and 200 scattered pilot carriers are distributed uniformly in the first and fourth symbol periods (3GPP, 2006). Let K be chosen as $K = N/2$, the SNR is 15dB.

As shown in Fig.13, the signal $u(k)$ has spectral power density of 10^{-3} outside the signal band. Compared with the original spectrum of $d(k)$ whose magnitude is 0 at the carriers for $|n| > N_1$, the information over the entire frequency band can be extracted from $u(k)$ though the spectral power density outside the band is a little lower than the inside part. It implies that it is possible to identify the dynamics of channel even outside the signal band.

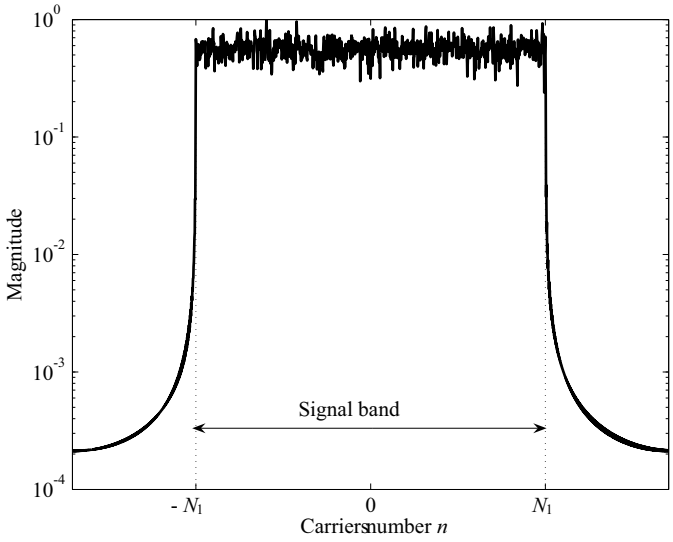


Fig. 13. Spectrum of $u(k)$

The true frequency property of $H(e^{jn\omega_0})$, where the longest effective delay tap $k_M = 100$ is used in channel estimation. The estimates after 50 iterations are plotted in Fig.14. It illustrates that the estimate is very close to the true one even outside the signal band, though the transmitted signal $d(k)$ has severe band limitation. As a comparison, the channel is also identified by conventional methods using RLS and LMS, and the results show that though

both RLS and LMS have estimated the channel property inside the signal band, they cannot provide satisfactory identification outside the signal band.

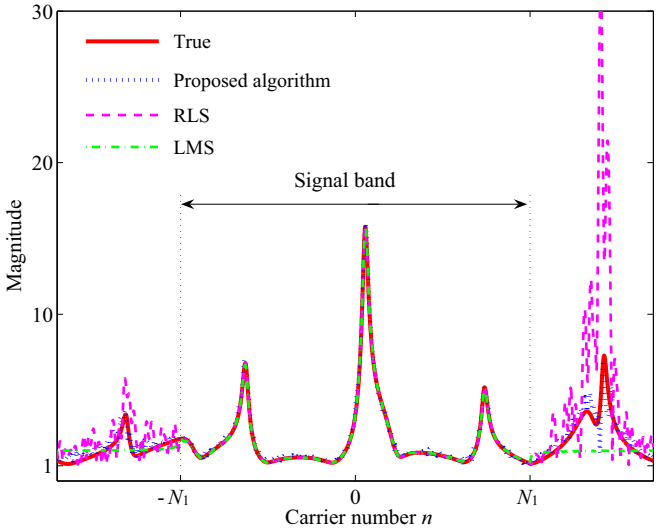


Fig. 14. Frequency property of communication channel

The estimation errors outside the signal band under various noise environments are illustrated in Fig.15, where the error is evaluated by

$$E_{H,\text{out}} = \frac{\sum\limits_{N_1<|n|<\frac{N}{2}} \left|\hat{H}(e^{jn\omega_0}) - H(e^{jn\omega_0})\right|^2}{\sum\limits_{N_1<|n|<\frac{N}{2}} \left|H(e^{jn\omega_0})\right|^2}. \tag{60}$$

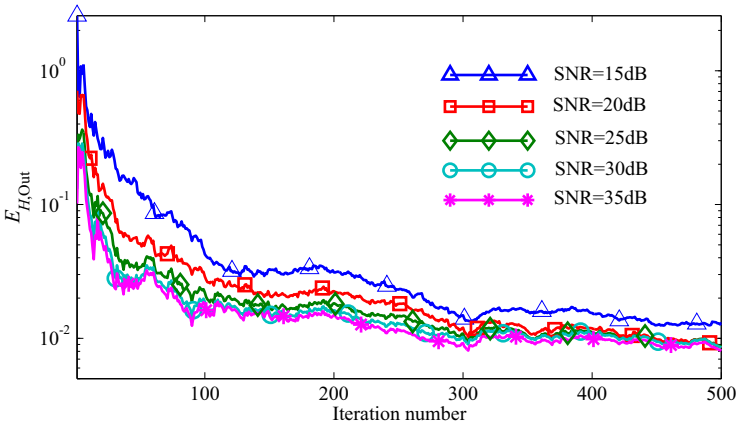


Fig. 15. Estimation error versus different SNR

It can be seen that even for low SNR, the estimation error successfully reduces to a low level just by tens iterations, and the estimated channel model can be applied to design adaptive filters for the OFDM system.

5. Conclusions

Channel identification using Fourier transform has been studied in this chapter. Since the OFDM transmitted signals in base band are generated through discrete Fourier transform, both the transmitted and received signals are easily managed in the frequency domain through Fourier transform. Consequently, the identification problem of OFDM channels could be solved in the frequency domain. Two channels have been investigated: the channel with long multipath interferences; and the transmitted signals with severe band limitation, where the conventional time domain methods cannot offer effective estimation. Firstly the properties of OFDM signals and the structural information have been analyzed in the frequency domain, then the relations of channel model and available information extracted from the observation data and structural information have been induced. Based on these relations, the frequency domain algorithms for OFDM channel identification have been developed, and the techniques have also been investigated to improve the identification accuracy, to deal with the time-varying channel and to reduce the computational complexity. It has been illustrated that the proposed frequency domain algorithms have better performance than the conventional time domain methods under the severe identification conditions considered in these problems, and the numerical results have demonstrated the effectiveness of Fourier transform in the channel identification applications. The algorithms work for the MIMO OFDM systems, and the estimation for OFDM channel with frequency offset are under further research work.

6. References

3GPP (2006). 3GPP TR 25.814 (release 7), *Technical report*, 3rd Generation Partnership Project. URL: *www.3gpp.org/ftp/Specs/html-info/TSG-WG–r1.htm*

Balakrishnan, J., Martin, R. & Johson, J. C. (2003). Blind, adaptive channel shortening by sum-squared auto-correlation minimization (SAM), *IEEE Trans. Signal Processing* 51(12): 3086–3093.

Burke, J., Zeidler, J. & Rao, B. (2005). CINR difference analysis of optimal combining versus maximal ratio combining, *IEEE Trans. Wireless Communications* 4(1): 1–5.

Chi, C., Feng, C., Chen, C. & Chen, C. (2006). *Blind Equalization and System Identification*, Springer.

Coleri, S., Ergen, M. & Bahai, A. (2002). Channel estimation techniques based on pilot arrangement in OFDM systems, *IEEE Trans. Broadcasting* 48(3): 223–229.

Ding, L., Zhou, T., Morgan, D., Ma, Z., Kenney, J., Kim, J. & Giardina, D. (2004). A robust digital baseband predistortion constructed using memory polynomials, *IEEE Trans. Commun.* pp. 159–165.

Ding, Z. & Li, Y. (2001). *Blind Equalization and Identification*, Marcel Dekker, Inc.

European Communities (1989). COST 207, digital land mobile radio communications, final report, *Technical report*, Commission of the European Communities.

Giannakis, G., Hua, Y., Stoica, P. & Tong, L. (2000). *Signal Processing Advances in Wireless and Mobile Communications,* Vol.1*: Trends in Channel Estimation and Equalization*, Prentice-Hall, Englewood Cliffs, NJ.

Glover, I. & Grant, P. (1998). *Digital Communications*, Prentice Hall.

Hamazumi, H., Imamura, K., Iai, N., Shibuya, K. & Sasaki, M. (2000). A study of a loop interference canceller for the relay stations in an SFN for digital terrestrial broadcasting, *GLOBECOM'00: IEEE Global Telecommunications Conference*, San Francisco, pp. 167–171.

Hamazumi, T. & Imamura, K. (2000). Coupling canceller, *Technical report*, NHK Science & Technology Research Laboratories.

Haykin, S. (2001). *Adaptive Filter Theory*, 4th edn, Prentice Hall.

Higuchi, K. & Sasaoka, H. (2004). Adaptive array suppressing inter symbol interference based on frequency spectrum in OFDM systems, *IEICE Trans. Commun.* J87-B: 1222–1229.

Hori, S., Kikuma, N. & Inagaki, N. (2003). MMSE adaptive array suppressing only multipath waves with delay times beyond the guard interval for fixed reception in the OFDM systems, *IEICE Trans. Commun.* J86-B: 1934–1940.

Koiveunen, V., Enescu, M. & Sirbu, M. (2004). Blind and semiblind channel estimation, *in* K. E. Barner & G. R. Arce (eds), *Nonlinear Signal Processing and Image Processing*, CRC Press, pp. 257–332.

Ljung, L. (1999). *System Identification – Theory for the User*, Prentice Hall, Upper Saddle River, NJ.

Muquet, M., Courville, M. & Duhamel, P. (2002). Subspace-based blind and semiblind channel estimation for OFDM systems, *IEEE Trans. Signal Processing* 50(7): 1699–1712.

Nguyen, V., Winkler, M., Hansen, C. & Kuchenbecker, H. (2003). Channel estimation for OFDM systems in case of insufficient guard interval length, *Proc. 15th International Conference on Wireless Communications*, Alberta, Canada.

Pintelon, R. & Schoukens, J. (2001). *System Identification–A Frequency Domain Approach*, IEEE Press.

Shibuya, K. (2006). Broadcast-wave relay technology for digital terrestrial television broadcasting, *Proceedings of the IEEE* 94(1): 269–273.

Sun, L. & Sano, A. (2005). Channel identification for SFN relay station with coupling wave in OFDM systems, *IEICE Trans. Fundamental* J88-A(9): 1045–1054.

Sun, L. & Sano, A. (2007). Channel identification and applications to OFDM communication systems with limited bandwidth, *Proc. 15th European Digital Signal Processing Conference*, Poznan, Poland.

Sun, L., Sano, A., Sun, W. & Kajiwara, H. (2009). Channel identification and interference compensation for OFDM system in long multipath environment, *Signal Processing* 89: 1589–1601.

Suzuki, N., Uehara, H. & Yokoyama, M. (2002). A new OFDM demodulation method with variable-length effective symbol and ICI canceller, *IEICE Trans. Fundamental* E85-A(12): 2859–2867.

Wang, X. & Poor, H. (2003). *Wireless Communication Systems- Advanced Techniques for Signal Reception*, Prentice Hall.

Ysebaert, G., Pisoni, F., Bonavetura, M., Hug, R. & Moonen, M. (2004). Echo cancellation in DMT-receivers: Circulant decomposition canceller, *IEEE Trans. Signal Processing* 52(9): 2612–2624.

Yu, J. & Su, Y. (2004). Pilot assisted ML frequency-offset estimation for OFDM systems, *IEEE Trans. on Communication* 52(11): 1997–2008.

5

Fast Fourier Transform Processors: Implementing FFT and IFFT Cores for OFDM Communication Systems

A. Cortés, I. Vélez, M. Turrillas and J. F. Sevillano
TECNUN (Universidad de Navarra) and CEIT
Spain

1. Introduction

The terms Fast Fourier Transform (FFT) and Inverse Fast Fourier Transform (IFFT) are used to denote efficient and fast algorithms to compute the Discrete Fourier Transform (DFT) and the Inverse Discrete Fourier Transform (IDFT) respectively. The FFT/IFFT is widely used in many digital signal processing applications and the efficient implementation of the FFT/IFFT is a topic of continuous research.

During the last years, communication systems based on Orthogonal Frequency Division Multiplexing (OFDM) have been an important driver for the research in FFT/IFFT algorithms and their implementation. OFDM is a bandwidth efficient multiple access scheme for digital communications (Engels, 2002; Nee & Prasad, 2000). Many of nowadays most important wireless communication systems use this OFDM technique: Digital Audio Broadcasting (DAB) (*World DAB Forum*, n.d.), Digital Video Broadcasting (DVB) (ETS, 2004), Wireless Local Area Network (WLAN) (IEE, 1999), Wireless Metropolitan Area Network (WMAN) (IEE, 2003) and Multi Band –OFDM Ultra Wide Band (MB–OFDM UWB) (ECM, 2005). Moreover, this technique is also employed in important wired applications such as Asymmetric Digital Subscriber Line (ADSL) or Power Line Communication (PLC).

OFDM systems rely on the IFFT for an efficient implementation of the signal modulation on the transmitter side, whereas the FFT is used for efficient demodulation of the received signal. The FFT/IFFT becomes one of the most critical modules in OFDM transceivers. In fact, the most computationally intensive parts of an OFDM system are the IFFT in the transmitter and the Viterbi decoder in the receiver (Maharatna et al., 2004). The FFT is the second computationally intensive part in the receiver. Therefore, the implementation of the FFT and IFFT must be optimized to achieve the required throughput with the minimum penalty in area and power consumption. The demanding requirements of modern OFDM transceivers lead, in many cases, to the implementation of special–purpose hardware for the most critical parts of the transceiver. Thus, it is common to find the FFT/IFFT implemented as aVery Large Scale Integrated (VLSI) circuit. The techniques applied to the FFT can be applied to the IFFT as well. Moreover, the IFFT can be easily obtained by manipulating the output of a FFT processor. Therefore, the discussion in this chapter concentrates on the FFT without loss of generality.

Different kinds of FFT algorithms can be found in the literature; e.g.: (Good, 1958; Thomas, 1963), (Cooley & Tukey, 1965), (Rader, 1968b), (Rader, 1968a), (Bruun, 1978) and (Winograd, 1978). Among the different kinds of FFT algorithms, the algorithms based on the approach proposed by James W. Cooley and John W. Tukey in (Cooley & Tukey, 1965) are very popular in OFDM systems. These Cooley–Tukey (CT) algorithms present a very regular structure, which facilitates an efficient implementation. The computations of the FFT is divided into $log_r(N)$ stages, where N is the number of points of the FFT and r is called the radix of the algorithm. Within each stage, data shuffling and the so-called butterfly computation and twiddle factor multiplications are performed. Usually, the butterfly operations are all identical and the twiddle factor multiplications and data shuffling follow some kind of pattern. This regular structure makes them very attractive for VLSI circuit implementation.

Different hardware architectures have been used in the literature for the implementation of the CT algorithms. The FFT hardware architectures can be classified into three groups:

- Monoprocessor: A single hardware element is used to perform all the butterflies, twiddle factor multiplications and data shuffling of each stage. The same hardware is reused for all the stages.
- Parallel: The computation of the butterflies, twiddle factor multiplications and data shuffling within one stage is accelerated by using several processing elements. The same hardware elements are again reused for all the stages.
- Pipeline: A single hardware element is used to perform all the butterflies, twiddle factor multiplications and data shuffling of each stage. However, in contrast to former categories, a different hardware element is used to process each stage.

It is common in the literature to further classify the pipeline architectures according to the structure used for the shuffling into two basic types: Delay Commutator (DC) and Delay Feedback (DF). Also, according to the number of lines of data used in these pipeline architectures, they can be classified into Single–path (S) or Multiple–path (M) architectures.

Many different variations of CT algorithms have been proposed in the literature to improve different aspects of the implementation (memory resources, number of arithmetic operations, etc.) and their mapping to a specific hardware architecture. Table 1 summarizes the features of some FFT/IFFT processors for OFDM systems proposed in the literature. Proposals such as (Chang & Park, 2004; Serrá et al., 2004) employed a monoprocessor architecture to process the FFT. (Jiang et al., 2004; Lin, Liu & Lee, 2004) used parallel architectures and (Kuo et al., 2003) chose a cached memory monoprocessor architecture for the FFT processing in an OFDM system. However, the most widely used architectures for the FFT/IFFT processor in an OFDM system are pipeline architectures. (Cortés et al., 2007; He & Torkelson, 1998; Lee & Park, 2007; Lee et al., 2006; Turrillas et al., 2010; Wang et al., 2005) propose Single-Path Delay Feedback (SDF) architectures. (Lin et al., 2005; Liu et al., 2007) employ an Multi-Path Delay Feedback (MDF) architecture. (Bidet et al., 1995) proposes a Single-Path Delay Commutator (SDC) architecture and (Jung et al., 2005; Saberinia, 2006) chose an Multi-Path Delay Commutator (MDC) architecture. (Saberinia, 2006) called the MDC architecture as Buffered Multi-Path Delay Commutator (BRMDC) due to the buffers used in the input data. Analyzing the radix, r, of the algorithms employed in the literature, it can be observed that:

- For the monoprocessor architectures, radix 2 (Serrá et al., 2004) and radix 4 (Chang & Park, 2004) algorithms have been used.

Reference	FFT points	Architecture	Algorithm	Application
(Jung et al., 2005)	64	Pipeline–MDC	r–2 DIT	WLAN
(Maharatna et al., 2004)	64	Pipeline	r–2 DIT	WLAN
(Serrá et al., 2004)	64	Monoprocessor	r–2 DIT	WLAN
(Lin et al., 2005)	128	Pipeline–MRMDF	MR $2/2^3$ DIF	UWB
(Saberinia, 2006)	128	Pipeline–BRMDC	r–2 DIF	UWB
(Lee et al., 2006)	128	Pipeline–SDF	r–2^4 DIF	UWB
(Liu et al., 2007)	64/128	Pipeline 8-path DF	MR DIF	UWB
(Cortés et al., 2007)	128	Pipeline	r–2^4 DIF	UWB
(Bidet et al., 1995)	8192	Pipeline–SDC	r–4/2 DIF	DVB–T
(Lin, Liu & Lee, 2004)	8192	Parallel	r–2^3 DIT	DVB–T
(Wang et al., 2005)	2/8 K	Pipeline–SDF	r–4/2 DIF	DVB–T
(Lenart & Owal, 2006)	2/4/8 K	Pipeline–SDF	r–2^2 DIF	DVB–T/H
(Lee & Park, 2007)	8 K	Pipeline–SDF	BD	DVB–T
(He & Torkelson, 1998)	1024	Pipeline–SDF	r–2^2 DIF	OFDM
(Kuo et al., 2003)	64–2048	Cached memory	r–2 DIT	OFDM
(Chang & Park, 2004)	1024	Monoprocessor	r–4 DIF	OFDM
(Jiang et al., 2004)	64	Parallel	r–2 DIT	OFDM
(Lin, Lin, Chen & Chang, 2004)	64	Pipeline	r–2 DIF	MIMO
(Rudagi et al., 2010)	64	Pipeline	r–2 DIT	OFDM
(Yu et al., 2011)	64	Pipeline	r–2 DIF	OFDM
(Tsai et al., 2011)	64	Pipeline	MR	OFDM
(Turrillas et al., 2010)	32 K	Pipeline	r–2^k DIF	DVB–T2

Table 1. FFT proposals for OFDM systems

- For the parallel architectures, FFT algorithms such as radix 2 (Jiang et al., 2004) and radix 2^3 (Lin, Liu & Lee, 2004) have been used.
- For the pipeline architectures, different FFT algorithms such as radix 2 (Jung et al., 2005; Rudagi et al., 2010; Saberinia, 2006; Yu et al., 2011), mixed radix (MR) (Bidet et al., 1995; Lin, Lin, Chen & Chang, 2004; Lin et al., 2005; Liu et al., 2007; Tsai et al., 2011; Wang et al., 2005), radix 2^2 (Cortés et al., 2007; He & Torkelson, 1998; Lenart & Owal, 2006; Turrillas et al., 2010), radix 2^3 (Cortés et al., 2007; He & Torkelson, 1998; Turrillas et al., 2010), radix 2^4 (Cortés et al., 2007; Lee et al., 2006; Turrillas et al., 2010) and balanced decomposition (BD) (Lee & Park, 2007) which reduces the number of twiddle factors have been proposed.

When the length of the FFT is not very large, the fixed-point format is the most widely used number representation format due to the area-saving with respect to the floating-point representation. Thus, (Chang & Park, 2004; Cortés et al., 2007; He & Torkelson, 1998; Jiang et al., 2004; Jung et al., 2005; Kuo et al., 2003; Lee et al., 2006; Lin, Lin, Chen & Chang, 2004; Lin et al., 2005; Liu et al., 2007; Maharatna et al., 2004; Saberinia, 2006; Serrá et al., 2004) proposed fixed–point FFTs for $N < 8192$. Nevertheless, (Wang et al., 2005) also used the fixed–point format for large FFTs. However, when the length of the FFT increases, different types of non fixed-point formats have been used. (Bidet et al., 1995) proposed a 8K processor that uses Convergent block floating–point to improve the quantization error of the FFT design. The block floating–point implementation or the semi–floating point (SFlP) format achieve a more efficient FFT design in (Lee & Park, 2007; Lenart & Owal, 2006). In a different approach, (Turrillas et al., 2010) employed a variable datapath technique to process a 32K points FFT.

From Table 1, it can be seen that many different algorithms and architectures have been proposed for OFDM systems. The designer must select the most appropriate algorithm and the most efficient architecture for that algorithm, given the specifications of a certain OFDM system. This selection is a difficult task. There is not a clear algorithm/architecture winner. Therefore, the designer should explore different algorithms and architectures in the literature to find the optimal one for the specific OFDM application under development.

The typical way of expressing the FFT algorithms in the literature is by means of summations or flow graph notation. Examples of these representations can be found in (He & Torkelson, 1998; Lee & Park, 2007; Lee et al., 2006; Lin, Lin, Chen & Chang, 2004; Lin, Liu & Lee, 2004; Lin et al., 2005; Liu et al., 2007; Maharatna et al., 2004; Tsai et al., 2006). These representations do not help the designer to understand the algorithm fast, therefore, making it difficult to relate it to its HW resources. Additionally, sometimes there is a lack of a general expression for the algorithm/architecture, which makes it harder to adapt and evaluate for a different OFDM system. Therefore, these representations are not practical for design space exploration. What the designer needs for efficient design space exploration is a general expression for the different design parameters of the algorithms which is easy to understand and makes it fast to map to hardware resources.

In (Pease, 1968), a matrix notation to express the FFT is proposed. Different FFT algorithms are obtained combining a reduced set of operators to simplify the implementation of parallel processing in a special–purpose machine. In (Sloate, 1974), H. Sloate used the same approach as (Pease, 1968) to demonstrate how several FFT algorithms previously defined could be derived using the matricial expressions. Additionally, he analyzed some new algorithms and worked out how to relate the matricial expressions to their implementation. However, that notation is not generalized for pipeline architectures. Recently, (Cortés et al., 2009) generalized the above approach presenting a unified approach for radix r^k pipeline SDC/SDF FFT architectures. Radix r^k pipeline FFT architectures are very efficient architectures that are well suited for OFDM systems.

This chapter reviews the matricial representation of radix r^k pipeline SDF FFT architectures. Thus, a general expression in terms of the FFT design parameters that can be linked easily to hardware implementation resources is presented. This way, the designer of FFT/IFFT processors for OFDM systems is provided with the tools for efficient design space exploration. The design space exploration and the optimal architecture selection procedure is illustrated by means of a case study. The case study analyzes the FFT/IFFT processor in a WLAN IEEE 802.11a transceiver. The high level analysis proposed in (Cortés et al., 2009) is extended to implementation level to select the most efficient FFT/IFFT core in terms of area and power consumption.

2. A unified matricial approach for radix r^k FFT SDF pipeline architectures

This section presents the matricial representation of radix r^k Decimation In Frequency (DIF) SDF pipeline architectures. First, the DFT matrix factorization procedure that leads to the FFT algorithms is reviewed. This review is used to define the basic types of matrices needed for the FFT. Next, in order to simplify the notation and latter mapping of the matricial representation to hardware resources, some operators are defined. Then, the general expression for radix r^k FFT SDF pipeline architectures is presented and the mapping to hardware resources illustrated.

2.1 Review of the DFT matrix factorization

The DFT can be expressed matricially as,

$$\mathrm{DFT}(\mathbf{x}) = \mathbf{T}_N \cdot \mathbf{x} \tag{1}$$

where

$$\mathbf{x}^T = \{\, x_0 \; x_1 \; x_2 \; \cdots \; x_{N-1} \,\}$$

and $\mathbf{T}_N$ is the size N square matrix of coefficients given by

$$(\mathbf{T}_N)_{mn} = e^{-j\frac{mn2\pi}{N}}. \tag{2}$$

$$\mathbf{T}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{-j\frac{2\pi}{N}} & e^{-j\frac{2\cdot 2\pi}{N}} & \cdots & e^{-j\frac{(N-1)2\pi}{N}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & e^{-j\frac{(N-1)2\pi}{N}} & e^{-j\frac{2(N-1)2\pi}{N}} & \cdots & e^{-j\frac{(N-1)^2 2\pi}{N}} \end{bmatrix}$$

In order to factorize $\mathbf{T}_N$, (Sloate, 1974) defines

$$\mathbf{T}_N = \mathbf{P}_N^{(r)} \cdot \mathbf{T}_N^{(p)}, \tag{3}$$

where r is the radix of the algorithm, $p = N/r$ is a positive integer ($p \in \mathbb{N}^*$) and $\mathbf{P}_N^{(r)}$ is the stride permutation matrix. This matrix, $\mathbf{P}_N^{(r)}$, is defined by its effect on a vector:

$$\mathbf{P}_N^{(r)} \cdot \{\, x_0 \; x_1 \; \cdots \; x_{N-1} \,\}^T = \{\, \mathbf{y}_0 \; \mathbf{y}_1 \; \cdots \; \mathbf{y}_{p-1} \,\}^T$$

with $\mathbf{y}_i = \{\, x_i \; x_{i+p} \; x_{i+2p} \; \cdots \; x_{i+(r-1)p} \,\}$. The matrix $\mathbf{T}_N^{(p)}$ can be expressed as

$$\mathbf{T}_N^{(p)} = [\mathbf{I}_r \otimes \mathbf{T}_{N/r}] \cdot \mathbf{D}_N^{(r)} \cdot [\mathbf{T}_r \otimes \mathbf{I}_{N/r}] \tag{4}$$

where $\mathbf{I}_z$ is the identity matrix of size z. The symbol $\otimes$ represents the Kronecker product. Given an $m \times n$ matrix $\mathbf{A}$ and a matrix $\mathbf{B}$, the Kronecker product is defined as

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{0,0}\mathbf{B} & a_{0,1}\mathbf{B} & \cdots & a_{0,n-1}\mathbf{B} \\ a_{1,0}\mathbf{B} & a_{1,1}\mathbf{B} & \cdots & a_{1,n-1}\mathbf{B} \\ \vdots & \vdots & & \vdots \\ a_{m-1,0}\mathbf{B} & a_{m-1,1}\mathbf{B} & \cdots & a_{m-1,n-1}\mathbf{B} \end{bmatrix}. \tag{5}$$

$\mathbf{D}_N^{(r)}$ is a diagonal matrix given by

$$\mathbf{D}_N^{(r)} = quasidiag(\{\, \mathbf{I}_p \; \mathbf{K}_p^1 \; \mathbf{K}_p^2 \; \cdots \; \mathbf{K}_p^{r-1} \,\}), \tag{6}$$

with $\mathbf{K}_p = diag(\{\, 1 \; e^{-j\frac{2\pi}{N}} \; e^{-j\frac{2\cdot 2\pi}{N}} \; \cdots \; e^{-j\frac{(p-1)2\pi}{N}} \,\})$. Replacing (4) in (3),

$$\mathbf{T}_N = \mathbf{P}_N^{(r)} \cdot [\mathbf{I}_r \otimes \mathbf{T}_{N/r}] \cdot \mathbf{D}_N^{(r)} \cdot [\mathbf{T}_r \otimes \mathbf{I}_{N/r}]. \tag{7}$$

Equation (7) shows how to write a DFT matrix in terms of smaller DFT matrices. This process can be repeated recursively until a expression in terms of $\mathbf{T}_r$ is arrived. Equation (7) is a matricial representation of the decomposition technique used in the well known Cooley-Tukey FFT (Cooley & Tukey, 1965).

2.2 Definition of operators

In order to simplify the notation, nine operators are defined. Two different types of operators can be distinguished: the reordering operators and the arithmetic operators.

- Reordering operators
  - *Shuffling Operator* $\mathbf{S}^{(a)}$: Let N be divisible by r^a,

$$\mathbf{S}^{(a)} = \mathbf{I}_{r^a} \otimes \mathbf{P}^{(r)}_{N/r^a}. \tag{8}$$

    Note that,

$$(\mathbf{S}^{(a)})^{-1} = \mathbf{I}_{r^a} \otimes (\mathbf{P}^{(r)}_{N/r^a})^{-1}. \tag{9}$$

- Arithmetic operators
  - *Butterfly Operator* $\mathbf{B}$: Let N be divisible by r,

$$\mathbf{B} = \mathbf{I}_{N/r} \otimes \mathbf{T}_r. \tag{10}$$

  - *First Twiddle Factor Multiplier Operator* $\mathbf{M1}^{(a,b)}$:
    Let N be divisible by r^{a+b},

$$\mathbf{M1}^{(a,b)} = \begin{cases} \mathbf{I}_{r^a} \otimes (\prod_{l=1}^{b}[\mathbf{P}^{(r)}_{r^l} \otimes \mathbf{I}_{N/r^{a+l}}] \cdot \mathbf{D}^{(r^b)}_{N/r^a} \\ \cdot \prod_{l=1}^{b}[\mathbf{P}^{(r)}_{r^l} \otimes \mathbf{I}_{N/r^{a+l}}]), & if r^b < \frac{N}{r^a} \\ \mathbf{I}_N, & if r^b \geq \frac{N}{r^a}. \end{cases} \tag{11}$$

  - *Second Twiddle Factor Multiplier Operator* $\mathbf{M2}^{(a)}$:
    Let N be divisible by r^{a+2},

$$\mathbf{M2}^{(a)} = \mathbf{I}_{r^a} \otimes \mathbf{D}^{(r)}_{r^2} \otimes \mathbf{I}_{N/r^{a+2}}. \tag{12}$$

  - *Third Twiddle Factor Multiplier Operator* $\mathbf{M3}^{(a,b)}$: Let N be divisible by r^{a+b} and $b \geq 2$,

$$\mathbf{M3}^{(a,b)} = \mathbf{I}_{r^a} \otimes ([\mathbf{P}^{(r)}_{r^2} \otimes \mathbf{I}_{r^{(b-2)}}] \cdot \mathbf{D}^{(r^2)}_{r^b} \cdot [\mathbf{P}^{(r)}_{r^2} \otimes \mathbf{I}_{r^{(b-2)}}]) \otimes \mathbf{I}_{N/r^{a+b}}. \tag{13}$$

2.3 Matricial representation of radix r^k pipeline FFT architecture

The decomposition procedure given in equation (7) can be applied recursively to devise the r^k SDF pipeline architectures. Let $N = r^{(kn_k+l)}$ with $\{k, n_k\} \in \mathbb{N}^*$ and $l \in \{0, 1, \ldots, k-1\}$, the matricial representation of r^k DIF SDF pipeline architectures is given by

$$\mathbf{T}_N = \begin{cases} \mathbf{Q}_N \cdot \mathbf{T}^{(p_k)}_N, & for \text{ l=0} \\ \mathbf{Q}_N \cdot \mathbf{H}^{(l,k,n_k)} \cdot \mathbf{T}^{(p_k)}_N, & for \text{ l>0} \end{cases} \tag{14}$$

$$\mathbf{T}^{(p_k)}_N = \prod_{m=0}^{n_k-1} \mathbf{H}^{(k,k,n_k-m-1)} \tag{15}$$

$$\mathbf{Q}_N = \prod_{i=0}^{n_1-1} \mathbf{S}^{(i)}, \tag{16}$$

where $p_k = r^{k(n_k-1)}$. The term $\mathbf{H}^{(b,k,i)}$ represents the i^{th} stage of a r^k FFT algorithm. When $b = 1$, $\mathbf{H}^{(1,k,i)}$ is given by,

$$\mathbf{H}^{(1,k,i)} = \mathbf{M1}^{(k\cdot i,k)} \cdot (\mathbf{S}^{(k\cdot i)})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i)}. \tag{17}$$

When $b = 2$, $\mathbf{H}^{(2,k,i)}$ reduces to,

$$\mathbf{H}^{(2,k,i)} = \mathbf{M1}^{(k\cdot i,k)} \cdot (\mathbf{S}^{(k\cdot i+1)})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i+1)} \cdot \mathbf{V}^{(2,k\cdot i)}, \tag{18}$$

where $\mathbf{V}^{(b,k\cdot i)}$ is given by

$$\mathbf{V}^{(b,k\cdot i)} = \mathbf{M2}^{(k\cdot i+b-2)} \cdot (\mathbf{S}^{(k\cdot i)+b-2})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i+b-2)}. \tag{19}$$

When b is even and $b > 2$, $\mathbf{H}^{(b,k,i)}$ is given by,

$$\mathbf{H}^{(b,k,i)} = \mathbf{M1}^{(k\cdot i,k)} \cdot (\mathbf{S}^{(k\cdot i+b-1)})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i+b-1)} \cdot \mathbf{V}^{(b,k\cdot i)} \cdot \left[\prod_{m=0}^{M=\frac{b}{2}-2} \mathbf{G}^{(b,k\cdot i,m)})\right], \tag{20}$$

where $\mathbf{V}^{(b,k\cdot i)}$ is given by (19) and $\mathbf{G}^{(b,k\cdot i,m)}$, by (22).

When b is odd and $b > 1$, $\mathbf{H}^{(b,k,i)}$ is given by,

$$\mathbf{H}^{(b,k,i)} = \mathbf{M1}^{(k\cdot i,k)} \cdot (\mathbf{S}^{(k\cdot i+b-1)})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i+b-1)} \cdot \left[\prod_{m=0}^{M=\frac{b-1}{2}-1} \mathbf{G}^{(b,k\cdot i,m)}\right], \tag{21}$$

where $\mathbf{G}^{(b,k\cdot i,m)}$ is given by (22).

$$\mathbf{G}^{(b,k\cdot i,m)} = \begin{cases} \mathbf{M3}^{(k\cdot i+2(M-m),b-2(M-m))} \\ \cdot(\mathbf{S}^{(k\cdot i+b-2m-3)})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i+b-2m-3)} \\ \cdot\mathbf{M2}^{(k\cdot i+b-2m-4)} \\ \cdot(\mathbf{S}^{(k\cdot i+b-2m-4)})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i+b-2m-4)}, \; if \text{ b even and b>2} \\ \mathbf{M3}^{(k\cdot i+2(M-m),b-2(M-m))} \\ \cdot(\mathbf{S}^{(k\cdot i+b-2m-2)})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i+b-2m-2)} \\ \cdot\mathbf{M2}^{(k\cdot i+b-2m-3)} \\ \cdot(\mathbf{S}^{(k\cdot i+b-2m-3)})^{-1} \cdot \mathbf{B} \cdot \mathbf{S}^{(k\cdot i+b-2m-3)}, \; if \text{ b odd and b>1 .} \end{cases} \tag{22}$$

When $N = r^{kn_k+l}$ with $l > 0$, (14) means that, after n_k stages with r^k structure, an additional processing stage with the structure of a r^l algorithm is required.

The above matricial representation is general in terms of N, r and k. The Decimation In Time (DIT) version of the architecture can be easily obtained by transposing the expressions.

2.4 Mapping to pipeline architectures

The structure of typical stages r^k pipeline SDF architecture is depicted in Figure 1. It can be observed in the first row of Figure 1 that the output of one stage is connected to the input of the next stage. Within each stage, there is a sequence of hardware processing elements which can perform the computations defined by the operators presented in Section 2.2 as illustrated in the second and third rows of Figure 1. The hardware required to implement the computations demanded by each operator is discussed in (Cortés et al., 2009). It is important to note that the shuffling operators surrounding the butterfly operators can be merged in the single device with feedback typical of the SDF architecture. To illustrate this, in the fourth row of Figure 1 the implementation of radix 2^k algorithms is presented.

Each stage $\mathbf{H}^{(k,k,i)}$ consists of k butterfly operators $\mathbf{B}$, with their corresponding shuffling and unshuffling $\mathbf{S}$ and $(\mathbf{S})^{-1}$ terms. The hardware that implements the arithmetic of the butterfly operators $\mathbf{B}$ is the same in all the stages of the FFT processor. The implementation of the butterfly depends on the value of r. The length of each of the delay lines used for the shuffling and unshuffling corresponding to the first butterfly of the first stage is N/r. The length of each of these delay lines is reduced by a factor of r from one butterfly to the next one as is shown in the fourth row of Figure 1. The number of delay commutator structures used for the shuffling and unshuffling around a butterfly unit is the same as the number of butterfly units.

Operators **M1**, **M2** and **M3** mainly translate to complex multipliers (C.M.) and Look-Up Tables (LUTs) to store the twiddle factors (T.F.). A stage of the form $\mathbf{H}^{(k,k,i)}$ has one twiddle factor multiplier operator **M1**. The number of twiddle factors stored in the LUT used to implement **M1** is N for the first stage and it reduces by a factor of r^k from one stage to the next. When $N = r^{kn_k}$, no complex multiplier is needed to implement the twiddle factor multiplications given by $\mathbf{M1}^{(k(n_k-1),k)}$ of the last stage $\mathbf{H}^{(k,k,n_k-1)}$, because $\mathbf{M1}^{(k(n_k-1),k)} = \mathbf{I}_N$. The final processing given by terms of the form $\mathbf{H}^{(l,k,n_k)}$ when $N = r^{kn_k+l}$, with $l \neq 0$, does not actually have a operator **M1**; i.e.: $\mathbf{M1}^{(kn_k,k)} = \mathbf{I}_N$.

For $k > 1$, twiddle factor multiplier operators **M2** appear. A stage $\mathbf{H}^{(b,k,i)}$ has $b/2$ operators **M2** when b is even and $(b-1)/2$ when b is odd. The number of twiddle factors stored in the LUT used to implement **M2** is always r^2.

For $k > 2$, twiddle factor multiplier operators **M3** appear. A stage $\mathbf{H}^{(b,k,i)}$ has $b/2 - 1$ operators **M3** when b is even and $(b-1)/2$ when b is odd. The number of twiddle factors stored in the LUT used for the implementation of the first **M3** within a stage $\mathbf{H}^{(b,k,i)}$ is r^b and it reduces by a factor of r^2 from one **M3** to the next one within the same stage.

2.5 Hardware resources of r^k pipeline architectures

In (Cortés et al., 2009), the complexity of the r^k algorithms in terms of area was analyzed. In this section, some of their conclusions are summarized.

For a given N, the designer can vary the parameters r and k to achieve different implementations. These parameters influence the reordering operators and the arithmetic operators.

The overall amount of memory words used for the delay lines due to the reordering operators depends neither on r nor k. In an SDF architecture, the amount of memory is $N - 1$ words.

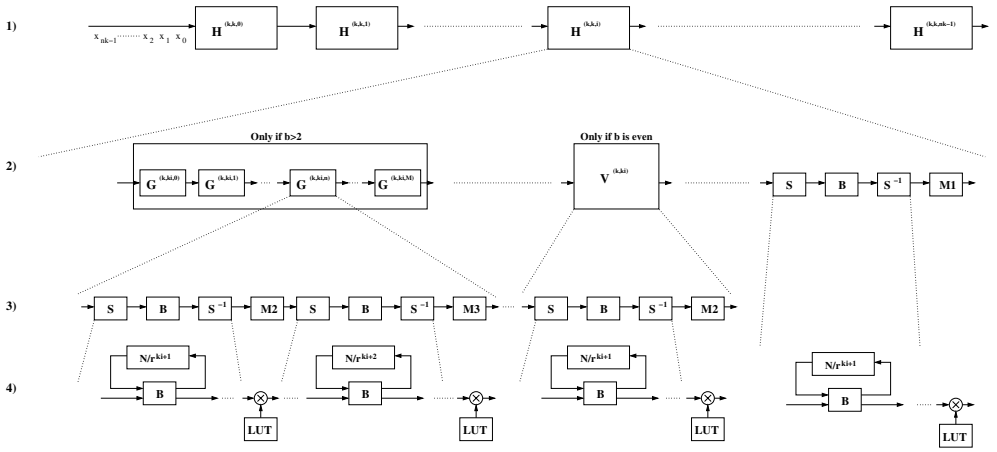


Fig. 1. Structure of typical stages of r^k pipeline SDF architectures

Therefore, the overall amount of memory words is fixed by the number of points of the FFT. $N-1$ is the minimum amount of memory reported in the literature for a pipeline FFT architecture. Modifying the values of r or k does not increase the amount of memory words used for the reordering operators.

Increasing the value of r for fixed values of N and k will reduce the number of butterflies and the overall number of complex multipliers. However, the butterflies become more complex. Regarding the number of twiddle factors, the number of twiddle factors due to **M1** are reduced, while the number of twiddle factors due to **M2** and **M3** increase. For small values of k, there is a reduction of the overall number of twiddle factors; while for large values of k, there is an increase in the overall number of twiddle factors. Thus, the benefit of increasing the value of r depends on the value of k.

If the values of N and r are fixed, the benefits of increasing k can be studied. The overall number of butterflies is $\log_r N$, and thus, increasing k does not increase either the number of butterflies or their area. Increasing the value of k reduces the number of complex multipliers due to **M1**. Once **M2** and **M3** appear in an architecture, the number of complex multipliers due to each one remains approximately constant. Thus, an important benefit of increasing the value of k is that it reduces the overall number of complex multipliers. When k increases, the number of twiddle factors due to **M1** is reduced. For $\lceil (\log_r N)/2 \rceil \leq k < \log_r N$, one single **M1** with N twiddle factors appears. Finally, when $k = \log_r N$, $\mathbf{M1} = \mathbf{I}_N$ and no hardware is needed to implement any **M1**. The number of twiddle factors due to **M2** remains constant with k. The number of twiddle factors due to **M3** increases with k. Adding the contributions of **M1**, **M2** and **M3**, the overall number of twiddle factors first reduces with k and then it starts to increase. It can be concluded that an optimum value of k exists for each value of N and r. Usually this optimum value is near $k = \lceil (\log_r N)/2 \rceil$.

From the general expressions, the designer has to look for those values of r and k that result in an optimum single-path FFT pipeline architecture for a given application. This search can be easily performed thanks to the close link between the proposed matricial representation and implementation. No other notation allows such an exploration within the pipeline architectures.

3. Case study: WLAN IEEE 802.11a

In this section, a case study is analyzed applying the proposed design space exploration in order to achieve the most efficient algorithm/architecture for an OFDM system. This design space exploration does not only analyze the hardware complexity of the algorithms as in (Cortés et al., 2009). Additionally, an implementation level analysis is carried out also taking into account the power consumption which is very important for wireless devices.

The main objective of this design space exploration is to select the most efficient radix r^k pipeline SDF FFT architecture for the OFDM application. An FFT for the WLAN IEEE 802.11a standard has been selected as the case study. In this case, the length of the FFT/IFFT is 64 points. According to (Cortés et al., 2009), the optimum value of k is near 3. Therefore, the following analysis concentrates on $r = 2^2$, $r = 2^3$ and $r = 2^4$ algorithms in order to perform the implementation level analysis and to study the silicon area and power consumption results.

The proposed design space exploration can be divided into four steps. At first, the OFDM system described by the standard is analyzed to extract the specifications. Once the OFDM specifications are known, the designer focuses on the system and on the FFT/IFFT specifications to determine the parameters needed for the FFT/IFFT design. The next step is the implementation level analysis of different FFT/IFFT algorithms and architectures in order to select the most efficient core for the WLAN system according to a given criterion. This step is composed of two main analysis: the Error Vector Magnitude (EVM) analysis in the transmitter and the Carrier-To-Noise Ratio (CNR) analysis in the receiver. After these analysis, the data bitwidth (*dbw*) and the twiddle factors bitwidth (*tbw*) are chosen so as not to degrade the system performance. Finally, the layout of the most efficient FFT/IFFT core for WLAN 802.11a in terms of area and power consumption is shown.

3.1 Study of the IEEE 802.11a standard

Table 2 shows the parameters specified in IEEE 802.11a standard for a WLAN system (IEE, 1999). $N_c = 64$ sub-carriers are used. $N_p = 4$ sub-carriers are used as pilot tones to make the coherent detection more robust against frequency offsets and phase noise. These pilot tones are always in sub-carriers -21, -7, 7 and 21 and they are modulated in Binary Phase-Shift Keying (BPSK). $N_d = 48$ tones are employed as data sub-carriers and the rest of sub-carriers are zero. The length of the guard interval must be longer than the delay spread of the channel.

$\mathbf{N}_c$	$\mathbf{N}_d$	$\mathbf{N}_p$	$\mathbf{N}_{gi}$	**R (Mbps)**	**BW (MHz)**	$\mathbf{t}_{sym}$ **(μs)**	**EVM (dB)**	$\mathbf{N}_f$	$\mathbf{L}_p$	H/N_o **(dB)**
64	48	4	16	54	20	4	-25	20	16	40

Table 2. OFDM Parameters for a WLAN IEEE 802.11a system

Considering an indoor environment, the necessary guard interval in a WLAN system is 0.8 μs. Therefore, the last $N_{gi} = 16$ data must be copied at the beginning of the OFDM symbol. The maximum data rate is $R = 54$ Mbps. Additionally, the bandwidth of the system is $BW = 20$ MHz. The standard also determines that the OFDM symbol period is $t_{sym} = 4\mu s$. In IEEE 802.11a, data can be modulated in BPSK, Quadrature Phase-Shift Keying (QPSK), 16-Quadrature Amplitude Modulation (QAM) or 64-QAM.

The EVM is defined as:

$$\text{EVM} = \sqrt{(I - I_o)^2 + (Q - Q_o)^2}, \tag{23}$$

where $I + jQ$ is the measured symbol and $I_o + jQ_o$ is the transmitted symbol. Figure 2 illustrates the computation of the EVM for a 16-QAM constellation. The IEEE 802.11a specifies

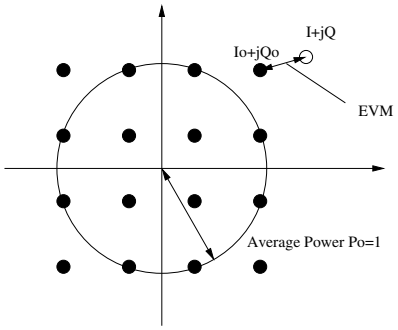


Fig. 2. EVM in the 16-QAM constellation

a mean value E_{rms} which has to be measured after transmitting $N_f = 20$ frames of $L_p = 16$ OFDM symbols with an average power P_o,

$$E_{\text{rms}} = \frac{1}{20} \sum_{i=1}^{20} \sqrt{\frac{\sum_{j=1}^{16} \left[\sum_{k=1}^{(48+4)} \text{EVM}(i,j,k) \right]}{(48+4)16 \cdot P_o}}, \tag{24}$$

where $\text{EVM}(i,j,k)$ is the magnitude of the error vector for the k^{th} sub–carrier of the j^{th} OFDM symbol of the i^{th} frame. The EVM is the quality constraint that must not be degraded in the transmitter. The EVM value in Table 2 is the minimum EVM in a WLAN system when the system transmits data modulated in 64–QAM with the maximum data rate of 54 Mbps.

The IEEE 802.11a standard specifies the spectral mask of the output signal in the transmitter. Figure 3 presents the spectral mask of the IEEE 802.11a transmitter output where this height is $H/N_o = 40dB$. The quality constraint selected for the receiver is the non–degradation of the CNR when the input signal to the receiver has the spectral mask specified by the standard. This is a first approach that produces an overconstrained core. EVM and H/N_o specifications are used to determine the allowed quantization error.

3.2 System analysis and FFT/IFFT specifications

The goal of this analysis is to determine the parameters of Table 3. Table 3 presents five important system specifications which influence significantly the design of the FFT/IFFT: the length of the FFT/IFFT N, the system clock frequency f_{clk}, the value of K_{tx} in the transmitter, the value of K_{AGC} and the CNR in the receiver. As the OFDM system supports 64 sub-carriers, the FFT core must be designed to perform 64-points FFT; i.e.: $N = 64$. As shown in Table 2, the OFDM symbol period (t_{sym}) in a WLAN system is equal to 4 μs. Following the analysis of (Velez, 2005), $f_{clk} = 60$ MHz is selected, higher than the maximum data rate R and multiple of the BW.

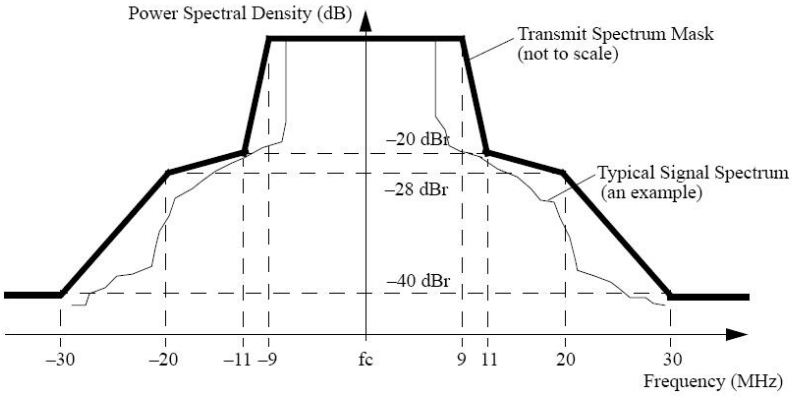


Fig. 3. Spectral mask of a transmitter output

N	Length of the FFT/IFFT
$\mathbf{f}_{clk}$	System clock frequency
$\mathbf{K}_{tx}$	Maximum gain to improve the DAC performance
$\mathbf{K}_{AGC}$	Maximum gain to improve the FFT performance
$\mathbf{CNR}_{max}$	Maximum carrier–to–Noise Ratio

Table 3. System specifications

3.2.1 k_{tx} in the transmitter

In order to select K_{tx}, an analysis of the IFFT output data must be carried out for each OFDM system. For this analysis, the modulation where the data reach the highest values in the I and Q data must be used.

In this analysis, K_{tx} is estimated by means of simulations. The transmission of the number of frames determined by the standard is simulated for different values of K_{tx} to calculate the EVM. Then, the largest K_{tx} that does not degrade the EVM can be selected. This way, the whole dynamic range of the Digital-to-Analog Converter (DAC) is exploited.

Figure 4 shows the system model used to analyze the EVM degradation with the increase of the value of K_{tx}. The system model is composed of a floating–point IFFT in the transmitter and a floating–point FFT in the receiver. The effect of the clipping in the DAC is emulated by a *limiter* located after the multiplication by K_{tx}. This *limiter* is responsible for saturating the amplified IFFT output according to the integer bits of the data representation. Two integer bits are considered for both the input and output data in the IFFT, since the maximum value of the input data is 1.08 when the modulation is 64–QAM. In order to improve the dynamic range of the IFFT output, the data are amplified by a factor K_{tx}. A factor K_{tx}, which is a power of two, is preferred to simplify the hardware implementation. In (Velez, 2005), the overflow probability of the different types of modulations in WLAN is analyzed carrying out Monte Carlo simulations with $3 \leq K_{tx} \leq 10$. It is concluded that BPSK modulation is the most sensitive to the clipping at the DAC. (Velez, 2005) also concluded that the most suitable value

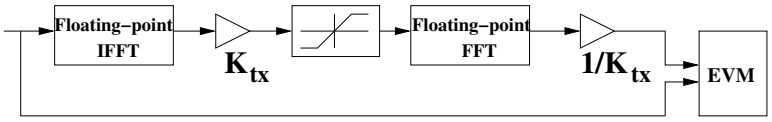


Fig. 4. System model to study the effect of K_{tx} in EVM

for K_{tx} is $K_{tx} = 4$ so as not to degrade the system performance. Additionally, $K_{tx} = 4$ is implemented as a simple shift in a real implementation.

Figure 5 shows the effect of the factor K_{tx} on the EVM in the transmitter, that is, how the EVM is degraded by the clipping produced in the IFFT. If a factor $K_{tx} = 8$ were selected, the

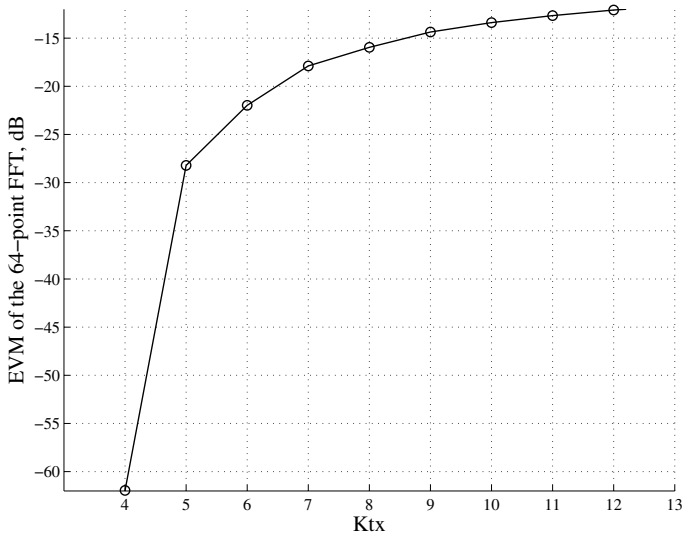


Fig. 5. Analysis of the effect of the K_{tx} in EVM

EVM value would not meet the standard specification. It can be observed that $K_{tx} = 4$ is a conservative decision, since the EVM specified by the standard for a BPSK modulation with a data rate of 9 Mbps is -8 dB, and the EVM obtained with $K_{tx} = 4$ is -61.95 dB.

3.2.2 k_{AGC} and CNR_{max} in the receiver

An Automatic Gain Control (AGC) module is a block found in many electronic devices. The AGC is commonly used to dynamically adjust the gain of the receiver amplifier to keep the received signal at the desired power level.

In this analysis, the effect of the AGC is modeled as a gain K_{AGC}. For the analysis, the value of K_{AGC} that does not degrade the CNR, is selected. This value is determined by means of simulations.

Figure 6 presents the system model to determine this factor K_{AGC}. This model is composed by a floating-point IFFT at the transmitter and a floating-point FFT at the receiver. The channel is modeled as as an Additive White Gaussian Noise (AWGN) channel.

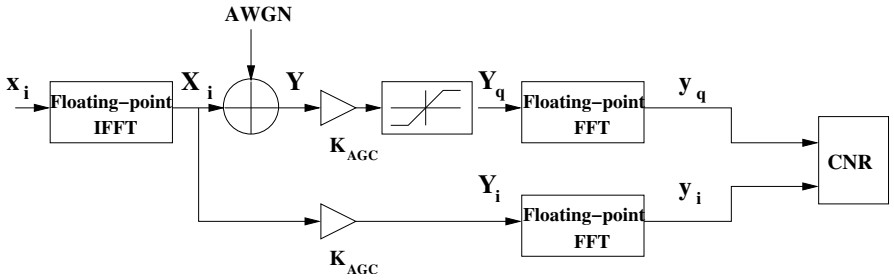


Fig. 6. System model to estimate the CNR for different values of K_{AGC}

Before applying the AWGN channel, the power of the output signal of the IFFT, X_i, is normalized to 1 dividing the signal by the factor λ, which is defined as

$$\lambda = \sqrt{\frac{N_d + N_p}{N_c^2}}. \tag{25}$$

In Figure 6, Y is the signal after applying the AWGN channel. At the receiver, this signal is multiplied by the gain K_{AGC} and a *limiter* is applied. This *limiter* is responsible for limiting the amplified signal according to the number of integer bits of the data representation. Once the *limiter* is applied, Y_q is the input of the floating-point FFT and y_q is the output of the floating-point FFT. y_i is the output of the floating-point FFT without noise and without applying the *limiter*. For the simulations, it is necessary to determine the variance of the noise to be applied to the transmitted signal so that the signal meets the spectral mask with the height H/N_o given by the standard. The Signal-To-Noise Ratio (SNR) of the transmitter output can be calculated from the height of the spectral mask H/N_o given by the standard. The power of the output signal of the transmitter can be written as,

$$P_o = \frac{(N_d + N_p) \cdot BW \cdot (H - N_o)}{N_c}, \tag{26}$$

where it is assumed that the power of the data sub-carriers is equal to the power of the pilot sub-carriers. The noise power is given by:

$$P_n = N_o \cdot BW. \tag{27}$$

Therefore,

$$\text{SNR} = \frac{P_o}{P_n} = (\frac{H}{N_o} - 1) \cdot \frac{(N_d + N_p)}{N_c}. \tag{28}$$

Then, the variance σ^2 of the noise in the channel is found to be,

$$\sigma^2 = \frac{1}{(\frac{H}{N_o} - 1) \cdot \frac{(N_d + N_p)}{N_c}}. \tag{29}$$

In order to select the value of the factor K_{AGC}, the CNR is used as a figure of merit. CNR is defined as,

$$\text{CNR} = \frac{\hat{P}_i}{\hat{P}_n}, \tag{30}$$

where $\hat{P}_i$ is the power of y_i and $\hat{P}_n$ is the power of the difference between y_i and y_q.

The gain K_{AGC} is selected so that the CNR is not degraded. Factor K_{AGC} fixes the power of the signal Y_q; i.e.: P_{Y_q}. The figures to analyze the effect of K_{AGC} on CNR plot the CNR versus P_{Y_q}, which is a more representative value.

The CNR is obtained by means of Monte Carlo simulations. This method allows the designer to estimate a measure of the performance of a communication system and the quality of the estimation itself (Sevillano, 2004). The looseness of our estimation in order to achieve a confidence interval equal to 0.95 is defined as,

$$\gamma = \frac{2 \cdot \sqrt{\hat{var}(C\hat{N}R)}}{C\hat{N}R} \tag{31}$$

Therefore, it can be said that the probability of CNR belonging to the interval $[(1-\gamma)C\hat{N}R, (1+\gamma)C\hat{N}R]$ is 0.95. The Monte Carlo simulations are stopped when $\gamma < 10^{-2}$.

It is assumed that the signal arrives with the maximum quality $H/N_o = 40$ dB. Therefore, applying (28), the SNR of the transmitted signal is of 39.09 dB.

Figure 7 shows the $(CNR)_{dB}$ versus the power of the input signal of the FFT P_{Y_q}. The signal works without degradation with $P_{Y_q} < -4$ dB, which corresponds to $K_{AGC} = 6$ in the simulations. For this value of K_{AGC}, $(CNR_{max})_{dB}$ is 39.1 dB.

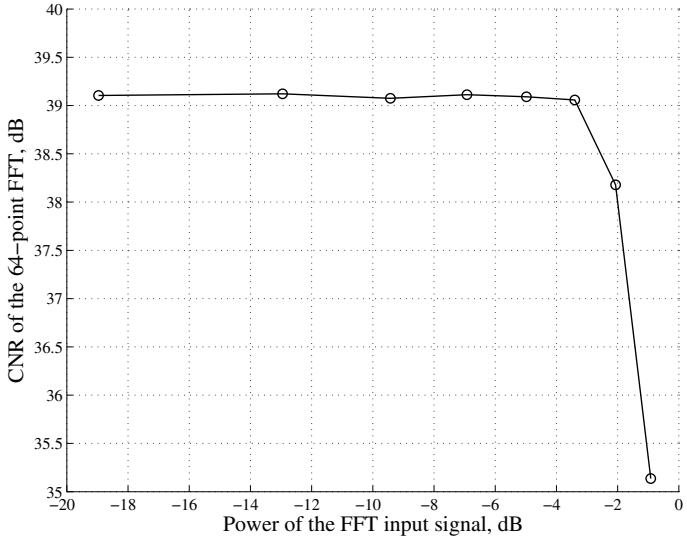


Fig. 7. $(CNR)_{dB}$ vs the power of the input signal in the FFT (P_{Y_q})

3.2.3 Summary of results

In this step, the length of the FFT N, the system clock frequency, f_{clk}, the gain factor in the transmitter, K_{tx}, and the gain factor in the AGC, K_{AGC} and the CNR_{max} have been estimated. Table 4 presents the necessary OFDM parameters for the next step.

N_{gi}	BW (MHz)	t_{sym} (μs)	EVM (dB)	N_f	L_p	f_{clk} (MHz)	N	K_{tx}	K_{AGC}	CNR_{max} (dB)
16	20	4	-25	20	16	60	64	4	6	39.1

Table 4. Specifications for the architecture selection for WLAN systems

3.3 Selection of algorithm/architectures

A high throughput FFT core is needed to fulfil the required specifications. Pipeline architectures are well suited to achieve small silicon area, high throughput, short processing time and reduced power consumption. Table 5 presents the hardware complexity of the candidate pipeline-SDF architectures. The pipeline-SDF radix 2^2 DIF architecture is composed

Architec.	r	Normal mults.	Constant mults.	ROM (regs.)	RAM (regs.)	adds /subs
pipeline–SDF	2^2	8	0	80	192	28
pipeline–SDF	2^3	4	4	64	192	40
pipeline–SDF	2^4	4	4	64	192	35

Table 5. Hardware complexity of candidate FFT architectures working at 60 MHz for WLAN systems

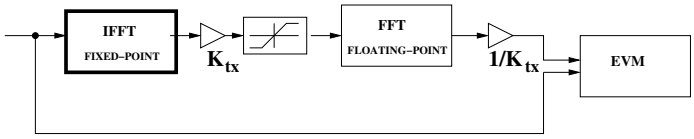
of six butterflies and two complex multipliers. The pipeline-SDF radix 2^3 DIF architecture is composed also of six butterflies, but one complex multiplier and two constants multipliers by one constant are used. The pipeline-SDF radix 2^4 DIF architecture is formed by six butterflies, one complex multiplier and one constant multiplier by two constants. Therefore, the three algorithms have the same number of multipliers taking into account normal and constant multipliers. The radix 2^3 and 2^4 DIF algorithms reduce ROM with respect to the radix 2^2 DIF one, but they add control logic.

To sum up, at this point it is not clear which is the most efficient architecture for WLAN. Therefore, it is necessary to make an implementation level analysis. The hardware complexity of radix 2^2, 2^3 and 2^4 DIF algorithms is compared to search the optimum design that meets the system specifications. The EVM constrains the word-length of the IFFT in the transmitter. The word-length in the receiver is constrained by the CNR. If the transmitter and the receiver are implemented in the same chip, the highest word-lengths must be chosen. The figures of merit to select the algorithm and architecture are the area and the power consumption estimated for an Application-Specific Integrated Circuit (ASIC) technology. Thus, this selection process can be stated as the problem of finding the FFT/IFFT processor which minimizes the AP cost function subjected to the constraints given by the specifications. The AP criterion trades off area and power consumption and can be used as a measure of the efficiency of the core. The area and power results presented in the following sections have been calculated for a TSMC 90 nm 6 ML technology with a clock frequency of 60 MHz working at 1.0V and a temperature of 25^{o}C. The area results have been estimated multiplying the cell area by a factor of 2.

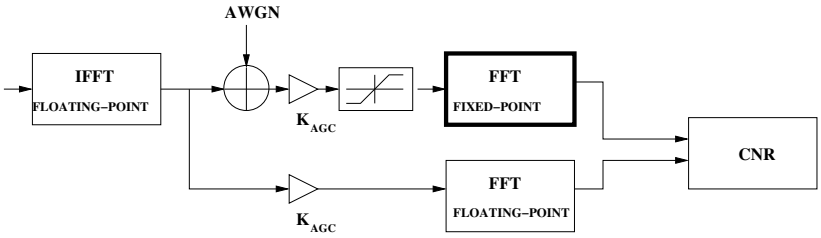
3.3.1 EVM analysis

In order to analyze the effect of the IFFT quantization error on the EVM during transmission, an ideal reception is considered. The system model is composed of a fixed-point IFFT at the

transmitter and a floating-point FFT at the receiver as can be seen in Figure 8(a). The EVM value is employed to select the values of *dbw* and *tbw* needed in the transmitter. An EVM margin of -10 dB is used in order to select a conservative word–length. Thus, a margin is left for other sources of error, such as the error produced by the analog processing. In order to



(a) EVM model



(b) CNR model

Fig. 8. System models to study the effect of quantization error on EVM and CNR

obtain the EVM results in Figures 9 and 10 the transmission of $N_f = 20$ frames of $L_p = 16$ OFDM symbols modulated in 64-QAM has been simulated. The pipeline-SDF with the radix 2^2, 2^3 and 2^4 DIF algorithms have been studied to find the most efficient implementation. For radix 2^3 and 2^4 algorithms, the bitwidth of the constant multiplier is assumed to be equal to the bitwidth of the normal multiplier.

Figure 9 shows the EVM and the area results of the pipeline–SDF DIF 64–point FFT/IFFT core for different algorithms. In order to guarantee an EVM of at least -35 dB, the radix 2^2 algorithm requires a (*dbw*,*tbw*) of (12,8), whereas the radix 2^4 and 2^3 algorithms need (12,7). For the given EVM, the radix 2^4 algorithm achieves the smallest core with $(dbw, tbw) = (12, 7)$.

The EVM and the power results of the pipeline–SDF DIF 64–point FFT/IFFT core are presented in Figure 10. It can be observed that the radix 2^2 algorithm requires less power consumption for the same (dbw, tbw) than the rest of algorithms, whereas the radix 2^3 algorithms consume more than the others.

Table 6 compares the area and the power consumption for the different algorithms to obtain an EVM of at least -35 dB. The power consumption in Table 6 has been normalized. (Kuo et al.,

Algorithm	**dbw**	**tbw**	**Area (mm^2)**	**P$_{norm}$**	**AP**
SDF r2^2	12	8	0.0993	0.0211	0.00210
SDF r2^3	12	7	0.0998	0.0233	0.00232
SDF r2^4	12	7	0.0960	0.0222	0.00214

Table 6. AP comparison for WLAN system with $EVM \leq -35dB$

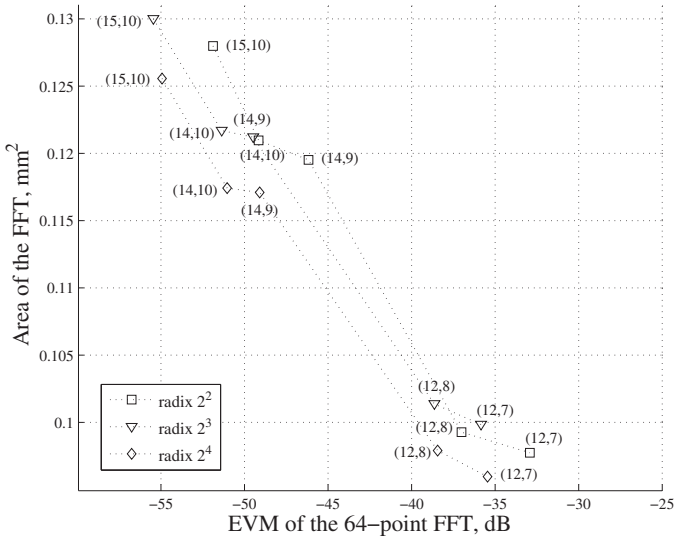


Fig. 9. EVM and area of the pipeline–SDF DIF 64–point FFT/IFFT core

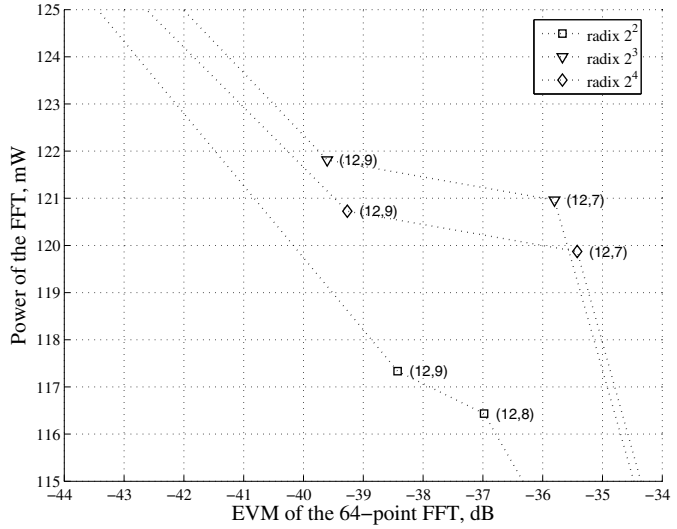


Fig. 10. EVM and power of the pipeline–SDF DIF 64–point FFT/IFFT core

2003) proposes a normalization of the power consumption as,

$$P_{norm} = \frac{P}{V_{DD}^2 \cdot N \cdot f_{clk}} \cdot 1000, \tag{32}$$

where P is the power consumption with a voltage of V_{DD} and working at f_{clk}. By slightly modifying the expression given by (Kuo et al., 2003) as follows,

$$P_{norm} = \frac{P}{V_{DD}^2 \cdot f_{clk}} \cdot 10^9. \tag{33}$$

The $r2^2$ algorithm needs larger bitwidths to achieve the target EVM. This extra bit in tbw increases the area needed. The $r2^3$ and $r2^4$ algorithms can achieve the required EVM with smaller tbw. The $r2^4$ algorithm with $dbw = 12$ and $tbw = 7$ achieves the most area–efficient implementation that fulfills the EVM specification. Nevertheless, the power consumption of the $r2^4$ algorithm is higher than the power consumption of $r2^2$ algorithm. In fact, the $r2^2$ algorithm is the most power–efficient design. In order to achieve a trade–off, the parameter $AP = Area \cdot P_{norm}$ is employed since it takes into account the area and the power consumption. The AP parameter trades off the area and power consumption of the core and, thus, it measures the efficiency of the design. For WLAN transmitter, it can be observed in Table 6 that the most efficient cores are the ones using $r2^2$ and $r2^4$ algorithms.

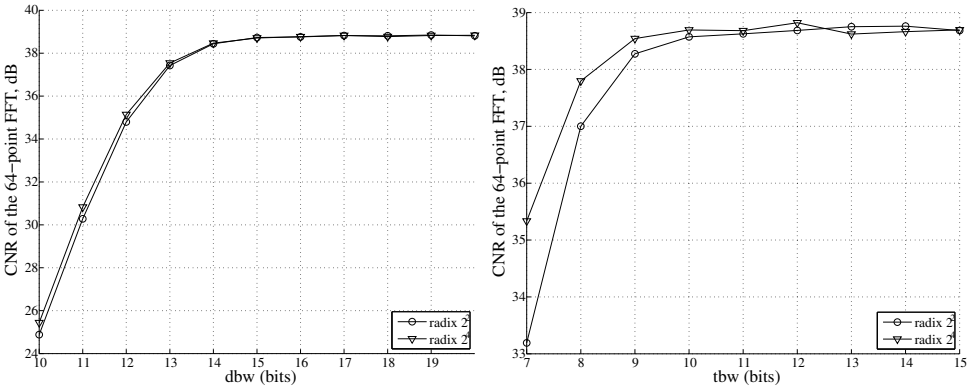
3.3.2 CNR analysis

After the EVM analysis, the $r2^3$ algorithm is discarded. Therefore, the CNR analysis focuses on the $r2^2$ and $r2^4$ algorithms. At this point, the area and power results of the FFT/IFFT core for different bitwidth configurations are already known. Then, the CNR analysis is used to select the dbw and tbw which fulfills the CNR specification.

In order to analyze the effect of the FFT quantization, the simulation model is formed by a floating–point IFFT at the transmitter and a fixed–point FFT at the receiver as is shown in Figure 8(b). First, only data are quantized. In this case, a figure of the CNR versus dbw can be used in order to select the dbw which does not degrade the CNR. Once dbw is selected, the twiddle factors are also quantized and a figure of the CNR versus tbw is shown in order to choose the tbw which does not degrade the CNR. This analysis is done for the candidates in order to determine the necessary dbw, tbw to comply with the CNR specification.

As an example, figures of $(CNR)_{dB}$ versus dbw and tbw are given for two algorithms. Figure 11(a) shows the $(CNR)_{dB}$ obtained versus the dbw parameter. In this case, the twiddle factors are not quantized. From the figure, a data bitwidth of $dbw = 15$ is selected to avoid degrading the $(CNR)_{dB}$ for both radix 2^2 and radix 2^4 algorithms. Once the data bitwidth is chosen, the twiddle factors are quantized. Figure 11(b) presents the $(CNR)_{dB}$ versus tbw where $dbw = 15$. In this case, a twiddle factor bitwidth of $tbw = 10$ is selected for both radix 2^2 and radix 2^4 algorithms. It can be observed that increasing more tbw does not improve the performance of the core.

Table 7 summarizes the (dbw, tbw) needed by the FFT to comply with the CNR requirement. Comparing the bitwidths needed by the IFFT in the transmitter to comply with the EVM and the ones needed by the FFT in the receiver to comply with the CNR, it can be said that the CNR is a much more restrictive specification. Therefore, the (dbw, tbw) selected for the FFT are the bitwidths used in the FFT/IFFT core. Table 7 presents the AP results of the FFT algorithms with the necessary bitwidths (dbw, tbw) to comply with the specifications. Taking into account the AP, the most efficient core for a WLAN system is the pipeline–SDF radix 2^2 DIF architecture.



(a) dbw (b) tbw ($dbw = 15$)

Fig. 11. Selection of dbw and tbw of the pipeline–SDF DIF 64–point FFT/IFFT core for radix 2^2 and radix 2^4 algorithms

Algorithm	dbw	tbw	Area (mm^2)	P_{norm}	AP
SDF r2^2	15	10	0.1284	0.0277	0.00356
SDF r2^4	15	10	0.1256	0.0287	0.00360

Table 7. AP comparison for WLAN system with $(CNR)_{dB} \geq 38.32$ dB

To sum up, the parameters of the final implementation of the pipeline–SDF radix 2^2 DIF architecture are shown in Table 8. As mentioned before, the system clock frequency is 60 MHz. Working at this frequency, the processing time of the chosen core is 0.67 μs. The data format is 2.13, whereas the format of the twiddle factors is 1.9. The EVM is -51.92 dB and the $(CNR)_{dB}$ is 38.57 dB. The estimated silicon area of the pipeline–SDF radix 2^2 DIF FFT/IFFT core is 0.1284 mm^2 and the power consumption estimation P_{norm} is 0.0277. It can be observed that the EVM complies with the specification in Table 4, and the CNR is close to the CNR_{max}.

N	dbw	tbw	f_{clk}	t_{proc}	CNR	EVM
64	15	10	60 MHz	0.67 μs	38.57 dB	-51.92 dB

Table 8. Core Parameters for the pipeline–SDF radix 2^2 DIF architecture

3.4 Layout of the FFT/IFFT core in an ASIC

Figure 12 shows the layout of the 64 complex–point FFT/IFFT fabricated in a 90 nm TSMC technology, 6–ML CMOS process. The core size is 0.1362 mm^2. In the previous section, the core area was estimated to be 0.1284 mm^2. It can be said that the area estimation is accurate enough. In order to present a comparison with the proposals found in the literature, the area of the cores is normalized as (B.M.Baas, 1999) using the equation:

$$A_{norm} = \frac{A}{(T_a/T_b)^2}, \tag{34}$$

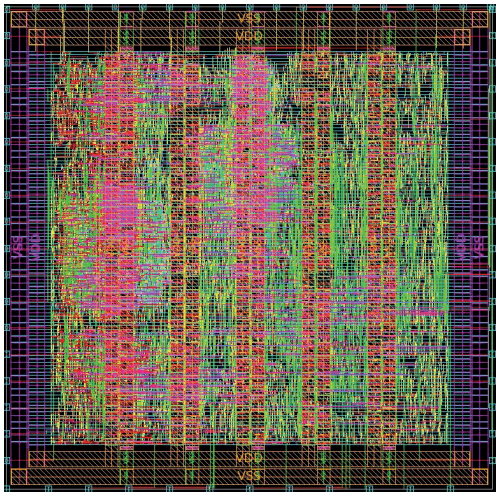


Fig. 12. Layout of the 64 complex–point FFT/IFFT fabricated in a 90 nm technology, 6–ML CMOS process. The core size is $0.370 \times 0.368\ mm^2$

.

where T_a is the anchor of the transistor of the technology actually used, A is the occupied area for T_a and T_b is the technology for which the area is normalized.

In order to assess the quality of the FFT/IFFT core, Table 9 makes a comparison with other 64 complex–point FFT/IFFT cores found in the literature. In Table 9, the area (A_{norm}) and the power (P_{norm}) have been normalized to a 90 nm technology using (34) and (33). The AP parameter indicates that our core is the most efficient one for a WLAN application. In fact, the presented core requires the smallest normalized power (P_{norm}).

	dbw	**tech.** (μ**m**)	$\mathbf{f}_{clk}$ **(MHz)**	$\mathbf{A}_{norm}$ **(mm^2)**	$\mathbf{t}_{proc}$ **(μs)**	$\mathbf{P}_{norm}$	**AP**
(Yu et al., 2011)	16	0.18	80	0.2209	–	0.0378	0.00835
(Tsai et al., 2011)	14.8	0.09	394	0.102	–	0.091	0.00928
Proposal	15	0.09	60	0.1362	0.67	0.0277	0.00377

Table 9. Comparison of fixed–point FFT processors for WLAN systems

4. Conclusions

Many different FFT/IFFT algorithms and architectures have been proposed in the literature for OFDM systems as has been presented in Section 1. Additionally, the usual FFT notations do not facilitate to perform a general analysis for the FFT/IFFT algorithm and architecture selection.

In Section 3, a design space exploration among different algorithms has been carried out. This search is hard to perform if general expressions are not available for the different algorithms in a unified way and if a mapping to the implementation can not be easily established. In this chapter, the matricial notation summarized in Section 2 is used as a tool to help the designer

in this search. The OFDM parameters obtained from the IEEE 802.11a standard analysis have been employed as constraints for the optimization problem. The AP parameter, which trades off area and power consumption, has been used as a measure of the efficiency of the core. Finally, a pipeline–SDF radix 2^2 DIF FFT/IFFT processor has been proposed, since it achieves the minimum of the AP cost function.

To sum up, it can be concluded that there is no unique FFT/IFFT algorithm, architecture and implementation that is optimal for all OFDM systems. Therefore, it is recommended to perform a search across the algorithm, architecture and implementation dimensions for each OFDM system. The matricial notation is presented in this chapter as a unified and compact representation that can help the designer in this search. This search is feasible, and the FFT/IFFT cores implemented using this approach present a great efficiency.

5. References

Bidet, E., Castelain, D., Joanblanq, C. & Senn, P. (1995). A Fast Single-Chip Implementation of 8192 Complex Point FFT, *IEEE Journal of Solid State Circuits* 30(3): 300–305.

B.M.Baas (1999). A Low-Power, High Performance, 1024-point FFT Processor, *IEEE Journal of Solid-State Circuits* 34(3): 380–387.

Bruun, G. (1978). z-Transform DFT Filters and FFTs, *IEEE Transactions on Acoustics, Speech and Signal Processing* 26(1): 56–63.

Chang, Y.-S. & Park, S.-C. (2004). An Enhanced Memory Assignment Scheme for Memory-Based FFT Processor, *IEEE Transactions Fundamentals* E87-A(11): 3020–3024.

Cooley, J. W. & Tukey, J. (1965). An Algorithm For Machine Calculation of Complex Fourier Series, *Mathematics of Computation* 19: 297–301.

Cortés, A., Vélez, I. & Sevillano, J. F. (2009). Radix r^k FFTs: Matricial Representation and SDC/SDF Pipeline Implementation, *IEEE Transactions on Signal Processing* 57(7): 2824–2839.

Cortés, A., Vélez, I., Irizar, A. & Sevillano, J. F. (2007). Area efficient IFFT/FFT core for MB-OFDM UWB, *Electronic Letters* 43(11).

ECM (2005). *Standard ECMA-368, High rate ultra wideband PHY and MAC standard*.

Engels, M. (ed.) (2002). *Wireless OFDM Systems. How to make them work?*, Kluwer Academic.

ETS (2004). *Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television*.

Good, I. J. (1958). The Interaction Algorithm and Practical Fourier Analysis, *Journal of the Royal Statistical Society* 20(2): 361–372.

He, S. & Torkelson, M. (1998). Designing Pipeline FFT Processor for OFDM (de)Modulation, *URSI International Symposium on Signals, Systems, and Electronics* 29: 257–262.

IEE (1999). *Part11: Wireless LAN Medium Access Control and Physical Layer specifications. High-speed Physical Layer in the 5 GHz Band*.

IEE (2003). *Part 16: Air interface for fixed broadband wireless access systems amendment 2: Medium Access Control modifications and additional physical layer specifications for 2-11 GHz*.

Jiang, M., Yang, B., Fu, Y., Jiang, A. & an Wang, X. (2004). Design Of FFT processor with Low Power Complex Multiplier for OFDM-based High-speed Wireless Applications, *International Symposium on Communications and Information Technologies* pp. 639–641.

Jung, Y., Yoon, H. & Kim, J. (2005). New Efficient FFT Algorithm and Pipeline Implementation Results for OFDM/DMT Applications, *IEEE Transactions on Consumer Electronics* 49(1): 14–20.

Kuo, J.-C., Wen, C.-H., Lin, C.-H. & Wu, A.-Y. (2003). VLSI Design of a Variable-Length FFT/IFFT Processor for OFDM-Based Communication Systems, *EURASIP Journal on Applied Signal Processing* pp. 1306–1316.

Lee, H.-Y. & Park, I.-C. (2007). Balanced Binary-Tree Decomposition for Area-Efficient Pipelined FFT Processing, *IEEE Transactions on Circuits and Systems-I* 54(4): 889–900.

Lee, J., Lee, H., Cho, S.-I. & Choi, S.-S. (2006). A High-speed, Low-complexity Radix-2 FFT Processor for MB-OFDM UWB Systems, *IEEE International Symposium on Circuits and Systems* pp. 4719–4722.

Lenart, T. & Owal, V. (2006). Architectures for Dynamic Data Scaling in 2/4/8k Pipeline FFT Cores, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 14(11): 1286–1290.

Lin, H.-L., Lin, H., Chen, Y.-C. & Chang, R. C. (2004). A Novel Pipelined Fast Fourier Transform Architecture for Double Rate OFDM Systems, *IEEE Workshop on Signal Processing Systems Design and Implementation* pp. 7–11.

Lin, Y.-W., Liu, H.-Y. & Lee, C.-Y. (2004). A Dynamic Scaling FFT Processor for DVB-T Applications, *IEEE Journal of Solid-State Circuits* 39(1): 2005–2013.

Lin, Y.-W., Liu, H.-Y. & Lee, C.-Y. (2005). A 1-GS/s FFT/IFFT Processor for UWB Applications, *IEEE Journal of Solid-State Circuits* 40(8): 1726–1735.

Liu, L., Ren, J., Wang, X. & Ye, F. (2007). Design of Low-Power, 1GS/s Throughput FFT Processor for MIMO-OFDM UWB Communication System, *IEEE International Symposium on Circuits and Systems* pp. 2594–2597.

Maharatna, K., Grass, E. & Jagdhold, U. (2004). A 64-Point Fourier Transform Chip for High-Speed Wireless LAN Application Using OFDM, *IEEE Journal of Solid-State Circuits* 39(3): 484–492.

Nee, R. V. & Prasad, R. (2000). *OFDM for Wireless Multimedia Communications*, Artech House.

Pease, M. (1968). An Adaptation of the Fast Fourier Transform for Parallel Processing, *Journal of the Association of Computing Machines* 15(2): 252–264.

Rader, C. M. (1968a). A Linear Filtering Approach to the Computation of the Discrete Fourier Transform, *Northeast Electronics Research and Engineering Meeting Record* 10: 218–219.

Rader, C. M. (1968b). Discrete Fourier transforms when the Number of Data Samples is Prime, *Proceedings of the IEEE* 56: 1107Ű–1108.

Rudagi, J. M., Lobo, R., Patil, P. & Biraj, N. (2010). An Efficient 64-point Pipelined FFT Engine, *2010 International Conference on Advances in Recent Technologies in Communication and Computing* pp. 204–208.

Saberinia, E. (2006). Implementation of a Multi-band Pulsed-OFDM Transceiver, *Journal of VLSI Signal Processing* 43: 73–88.

Serrá, M., Martí, P. & Carrabina, J. (2004). IFFT/FFT core architecture with an Identical Stage Structure for Wireless LAN Communications, *IEEE 5th Workshop on Signal Processing Advances in Wireless Communications* pp. 606–610.

Sevillano, J. F. (2004). *Diseño de un Demodulador QPSK para Sistemas de Radiodifusión Digital por Satélite*, PhD thesis, Campus tecnológico de la Universidad de Navarra.

Sloate, H. (1974). Matrix Representation for Sorting and the FFT, *IEEE Transactions on Circuits and Systems* CAS-21(1): 109–116.

Thomas, L. H. (1963). Using a Computer to Solve Problems in Physics, *Applications of Digital Computers* .

Tsai, P.-Y., Chen, C.-W. & Huang, M.-Y. (2011). Automatic IP Generation of FFT/IFFT Processors with Word-length Optimization for MIMO-OFDM Systems, *EURASIP Journal on Advances in Signal Processing* 2011(ID-136319).

Tsai, T. H., Peng, C. C. & Chen, T.-M. (2006). Design of a FFT/IFFT Soft IP Generator using on OFDM Communication System, *WSEAS Transactions on Circuits and Systems* 5(8): 1173–1180.

Turrillas, M., Cortés, A., Sevillano, J. F., Vélez, I., Oria, C., Irizar, A. & Baena, V. (2010). Comparison of area–efficient FFT algorithms for DVB-T2 receivers, *Electronic Letters* 46(15).

Velez, I. (2005). *Metodología de Diseño de Sistemas Digitales para Telecomunicaciones basada en C/C++: Aplicación a WLAN 802.11a*, PhD thesis, Campus tecnológico de la Universidad de Navarra.

Wang, C. C., Huang, J. M. & Cheng, H. C. (2005). A 2K/8K mode Small-Area FFT Processor for OFDM Demodulation of DVB-T Receivers, *IEEE Transactions on Consumer Electronics* pp. 28–32.

Winograd, S. (1978). On Computing the Discrete Fourier Transform, *Mathematics of Computation* 32: 175–199.

World DAB Forum (n.d.). `http://www.worlddab.org/`.

Yu, C., Yen, M.-H. & Hsiung, P.-A. (2011). A Low-Power 64-point Pipeline FFT/IFFT Processor for OFDM Applications, *IEEE Transactions on Consumer Electronics* 57(1): 40–45.

6

FPGA Implementation of Inverse Fast Fourier Transform in Orthogonal Frequency Division Multiplexing Systems

Somayeh Mohammady, Nasri Sulaiman,
Roslina M. Sidek, Pooria Varahram, M. Nizar Hamidon
Universiti Putra Malaysia (UPM)
Malaysia

1. Introduction

In modern communication systems, Orthogonal Frequency Division Multiplexing (OFDM) systems are used to transmit with higher data rate and avoid Inter Symbol Interference (ISI). The OFDM transmitter and receiver contain Inverse Fast Fourier Transform (IFFT) and Fast Fourier Transform (FFT), respectively. The IFFT block provides orthogonality between adjacent subcarriers. The orthogonality makes the signal frame relatively secure to the fading caused by natural multipath environment. As a result OFDM system has become very popular in modern telecommunication systems. Beside all the advantages of OFDM system, there is a main drawback of high Peak to Average Power Ratio (PAPR). There have been many approaches on reducing PAPR in time domain and frequency domain. Some of them work in time domain such as Partial Transmit Sequence Insertion (PTS) and some other methods perform in frequency domain such as Dummy Sequence Insertion (DSI) and Selected Mapping (SLM) methods (Bauml et al., 1996; Muller et al., 1997). Since according to (Baxley et al., 2007), the SLM method reduce PAPR with the least computational complexity and least additional modification requirements on the current technology, therefore most of recent researches have considered SLM based method modifications for their work. Most of these methods modified the OFDM transmitter in a way that multiple IFFT processors are required for implementation. This will increase the number of additions and multiplications that are needed for implementation.

In this Chapter, the OFDM system and the main block of IFFT are introduced. The IFFT block is implemented on FPGA and verification results are discussed. The Optimum Phase Sequence Insertion with Dummy Insertion (OPS-DSI) method is one of recent PAPR reduction techniques and a good example of application for IFFT processor is studied in this Chapter and the FPGA implementation result is verified with simulation results.

2. OFDM system

Fig. 1 shows how OFDM signal is processed. The data input signal with high data rate is split into narrow band channels with lower data rate and then, they are modulated by using general signal modulation (PSK, QAM) and followed by with Inverse Fast Fourier

Transform (IFFT) which provides orthogonality between adjacent sub-channels. After IFFT, the last portion of signal is copied to the head to provide immunity to Inter Symbol Interference (ISI) which is shown by Cyclic Prefix (CP) in Fig. 1.

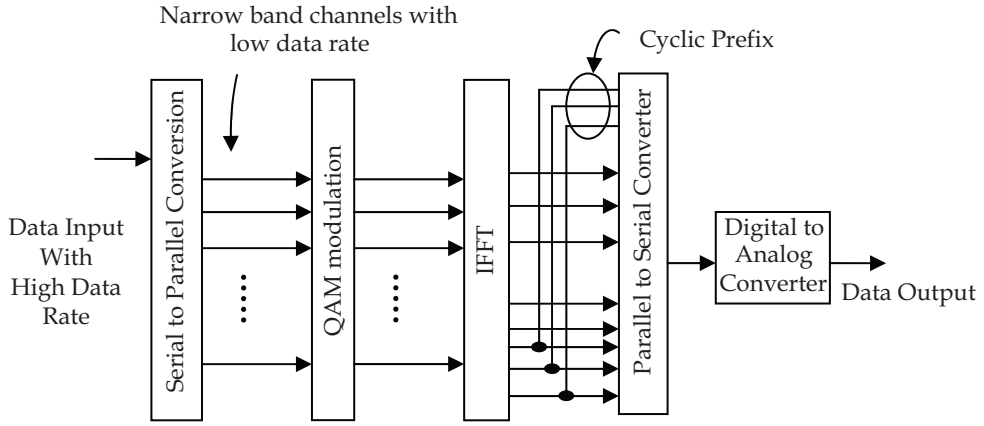


Fig. 1. The OFDM signal structure

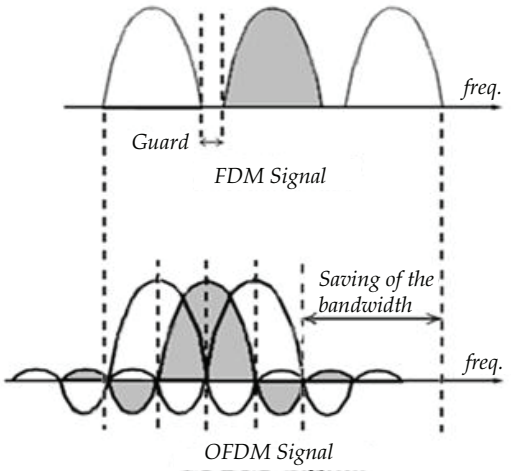


Fig. 2. Comparison between FDM and OFDM signals

Although OFDM process is similar to Frequency Division Multiplexing (FDM) signal but, there are some differences. FDM is a single carrier signal in which signal is divided into frequency bands with some guard interval between them to avoid interferences. This is not an issue in OFDM signal since neighboring subcarriers are orthogonal to each other; overlapping does not create interference and the bandwidth is used more efficiently as shown in Fig. 2.

The OFDM is a form of Frequency Division Multiplexing (FDM) scheme and Multicarrier Modulation (MCM) scheme. In FDM, each signal is divided to smaller signals named

subchannel with different frequency widths. Each subcarrier carries a signal at the same time in parallel. Every subcarrier is modulated by a constellation symbol.

3. IFFT implementation on Field Programmable Gate Array (FPGA)

Field Programmable Gate Arrays (FPGAs) are configurable and re-programmable digital logic devices, and programming code is usually written in Hardware Description Languages (HDL).

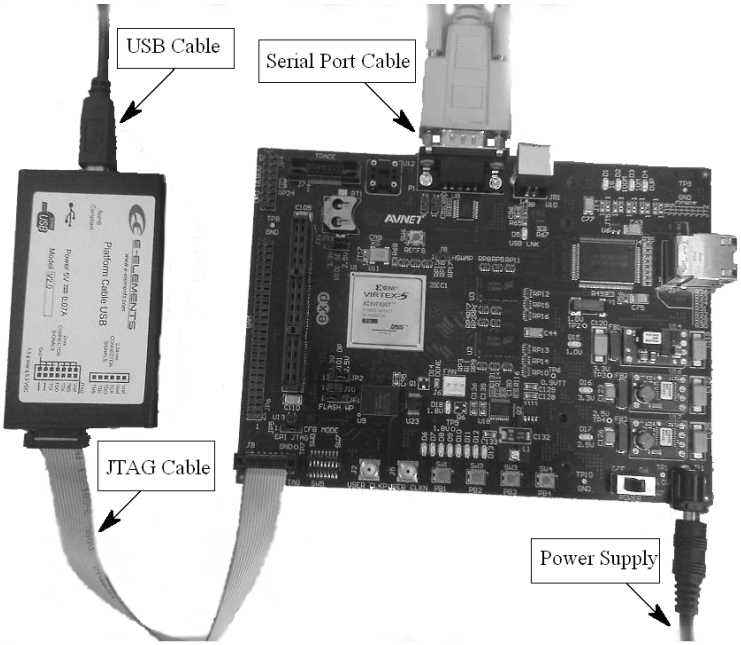


Fig. 3. Virtex-5 Pro development board

Virtex-5 FXT Evaluation Kit is used for implementation which has the Xilinx Virtex-5 XC5VFX30T-FF665 FPGA chip. The datasheet of this FPGA is provided in Appendix D. This board also has 64 MB DDR2 SDRAM memory and 16 MB FLASH memory with variety of I/O gates which makes it suitable for a typical implementation.

As shown in Fig. 3, the JTAG-USB cable is connected to the JTAG connector of the FPGA board for programming the FPGA. This cable should be connected to the USB port of the PC. The other connector is named serial port which is used to command while the program is performing.

This section introduces fundamentals of IFFT block prototype at the transmitter and the FFT block at the receiver. The basic equations of the FFT and the Inverse FFT (IFFT) are given by:

$$X(k)=\sum_{n=1}^{N-1} x(n)e^{-j2\pi kn/N} \quad ,k=0,...,N-1 \tag{1}$$

$$x(n)=\frac{1}{N}\sum_{n=1}^{N-1}X(k)e^{-j2\pi kn/N} \quad ,n=0,...,N-1 \tag{2}$$

where N is the transform size or the number of sample points in the data frame and $j=\sqrt{-1}$. $X(k)$ is the frequency output of the FFT at k^{th} point where k=0, 1, …, N-1 and $x(n)$ is the time sample at n^{th} point with n=0, 1,…, N-1.

Due to the symmetric of the exponential matrix $e^{-j2\pi kn/N}$, it can be represented as twiddle factor that is shown with ${W_N}^{nk}$. The computation can be performed faster by using twiddle factor as it depends on the number of points used and there is no need to recalculate it and the values can be referred to a matrix of twiddle factors. As the transform time is very crucial in FFT process, there is always a trade-off between the core size and the transform time. In Xilinx there are four architectures of Pipelined-Streaming I/O, Radix4-Burst I/O, Radix2-Burst I/O and Radix2-Lite-Burst I/O. They have different features to cover different time and size requirements.

In Pipelined Streaming I/O architecture, the data is processed continuously. The Radix4 uses an iterative approach to process the data. The data is loaded and processed separately. It is smaller in size than the pipelined solution howeverhas a longer transform time. The third architecture has the same iterative approach as Radix4 althoughthas longer transform time. The Radix2 is based on Decimation In Frequency (DIF) and separates the input data into two halves of:

$$X(0), X(1),\ldots, X(\frac{N}{2}-1) \, and \, X(\frac{N}{2}), X(\frac{N}{2}+1),\ldots, X(N-1) \tag{3}$$

The FFT formula for both even and odd conditions can be written in two summations as follows:

$$X(k)=\sum_{n=0}^{\frac{N}{2}-1} a(n)W_{\frac{N}{2}}^{nk} \, , where \, a(n)=x(n)+x(n+\frac{N}{2})$$

$$and \tag{4}$$

$$X(2k+1)=\sum_{n=0}^{\frac{N}{2}-1} b(n)W_{\frac{N}{2}}^{nk} \, , where \, b(n)=x(n)-x(n+\frac{N}{2})$$

This operation for 2 points can be graphically presented in Fig. 4.

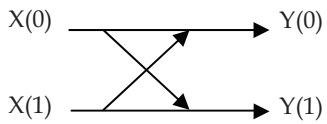


Fig. 4. Two point butterfly graph

where $Y(0)=X(0)+X(1)$ and $Y(0)=X(0)-X(1)$, respectively. This FFT flow graph is called butterfly graph. When the number of point is increased, the butterfly is expanded. Radix2 scheme in third architecture separates the input data into two halves therefore the butterfly

is smaller. This means it is smaller in size than the Radix4 solution. The forth scheme is based on the Radix2 architecture. The "Radix2-Lite-Burst I/O" uses a time-multiplexed approach to the butterfly and the butterfly is even smaller howeverthe transform time is longer. In this project the Radix2-Burst I/O architecture as shown in Figure 5.4, is used due to the less hardware resource requirement compared to the other algorithms to prototype FFT (Yiqun et al., 2006).

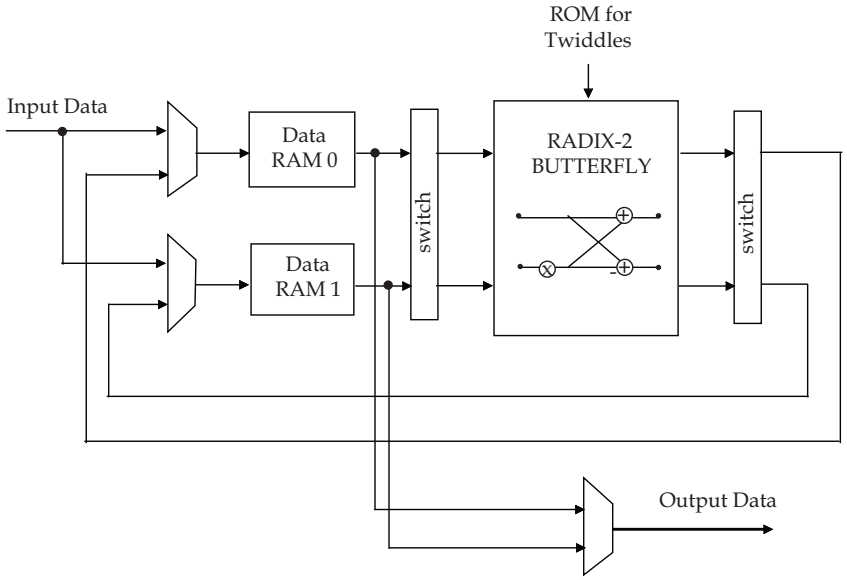


Fig. 5. Block diagram of the Radix2-Burst I/O architecture (Hemphill et al., 2007)

The signal cannot be simultaneously loaded and unloaded like Radix4, Burst I/O architecture and the loading should be stopped during the calculation of the transform.

The point sizes can be from 8 to 65536 and a minimum of block memories is used in this algorithm. When the point size is equal or less than 1024, both block memory and distributed memory can be used for data memories and phase memories.

In order to have accurate IFFT block, the model of targeted FPGA should be indicated in AccelDSP tool window. The AccelDSP is a synthesis tool that transforms a design in Matlab into a hardware module. This module can be VHDL or Verilog code. This tool controls an integrated environment with other design tools such as Matlab and Xilinx ISE tools.

There is a browser in GUI that shows the design hierarchy, the M-files, and the generated HDL source files. In this project, AccelDSP is used to generate the IFFT and FFT blocks. To guide the synthesis process, the design objects in the project explorer window is used. There are some parameters that should be defined here.

One of the important parameters in design of IFFT block is the algorithm to implement it which was discussed before and it is selected the Radix. The other parameter is the IFFT length that denotes the number of differential points in the IFFT. There is also option for I/O

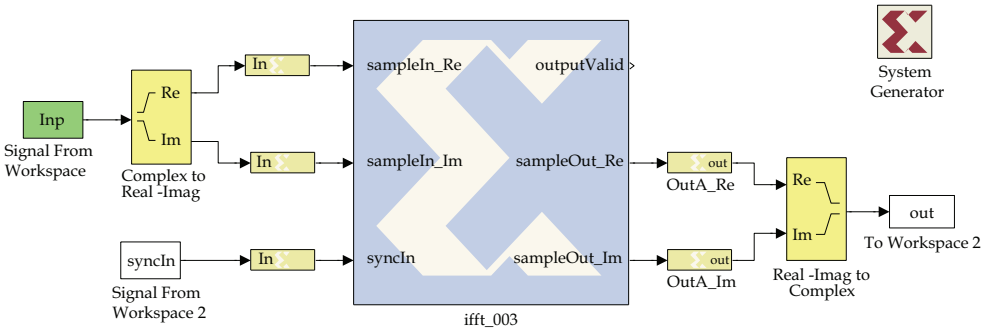


Fig. 6. System Generator block diagram of IFFT block and input output blocks

format. With the data I/O format option in AccelDSP GUI, input and output data can be initialized. Single buffering does not parallel any operations. Double buffering parallels the loading and unloading of frames of data. Natural Order I/O only applies to Single Fly architectures. Decimation Algorithm will naturally have inputs or outputs in digit/bit reverse ordering; DIF has natural order input and digit/bit reverse output, DIT has digit/bit reverse input and natural order output. The 'Yes' will force input and output to be natural order regardless of decimation type. The input data can be set to complex or real. Decimation algorithm parameter will be set to Decimation In Time (DIT) or Decimation In Frequency (DIF) algorithm. Scaling is the 1/IFFT Length ratio that can be set. Complex multiplier is another option that chooses different complex multiplier architectures. Round Mode sets the Quantizer round mode property for all data path quantizers. If Floor is selected for round mode, the numbers between 0 and -1 will be rounded to 0 and all the other numbers, bigger than zero and lower than -1, will round to the closest number. For example -1.8 will be -2. There is also a section for input data width that shows the number of bits used to represent the input. Input Data Fract Width shows the number of bits used to represent fractional part of input word width. Twiddle width is another parameter that shows the number of bits used to represent twiddle factors. In addition twiddle factor width is the number of bits used to represent the fractional part of phase factors. The range of the phase factors is (-1, 1) and therefore 2 bits are always needed for the integer part of the phase factors. The fractional part will always, twiddle factor width=twiddle width - 2. The data width can be also modified for output of IFFT. Output data width depends on the scaling option. If Scaling is set to 'Yes', output data width = input data width. If scaling is set to 'No", output data width = input data width + *log*2 (IFFT length) + 1. The output data fractional width indicates the greater of input data fractional width or twiddle fractional width.

Another important setting in AccelDSP tool is about the form of flow in the design. For this particular application, the flow should be set to System Generator. At the end of design, a library including the IFFT block with desired name is created. From Matlab simulink environment the library and IFFT block is accessible.

When the IFFT block is inserted in simulink window, some other components are required to complete the model which is shown in Fig. 6. These components are Input data, signal Synchronization line, output gate, and complex to real imaginary converter. At this time the

model should be able to run successfully. As shown in Fig. 6, the centre block with Xilinx sign at the background is the IFFT model which is generated by AccelDSP tools.

Then with system generator block, the NGC Netlist file can be generated. This file contains information that the ISE software is able to analyze it and estimate the hardware resource consumption which is presented in a table in ISE.

4. Results and discussions

When the IFFT block is designed in AccelDSP tool, the fixed point model of the design is generated. The AccelDSP automatically runs a MATLAB fixed-point simulation. Then the verification process can be done visually which is to compare the Fixed-Point Plot with the Floating-Point Plot to verify a match.

Fig. 7 presents an amplitude comparison between generated fixed point model of IFFT design with N=256 and Radix 2 shown by (a) and floating point model of this design which is shown by (b). The x-axis unit is the Number of samples per time (N). It can be observed that these two results are the same and therefore the IFFT processor is verified in terms of amplitude.

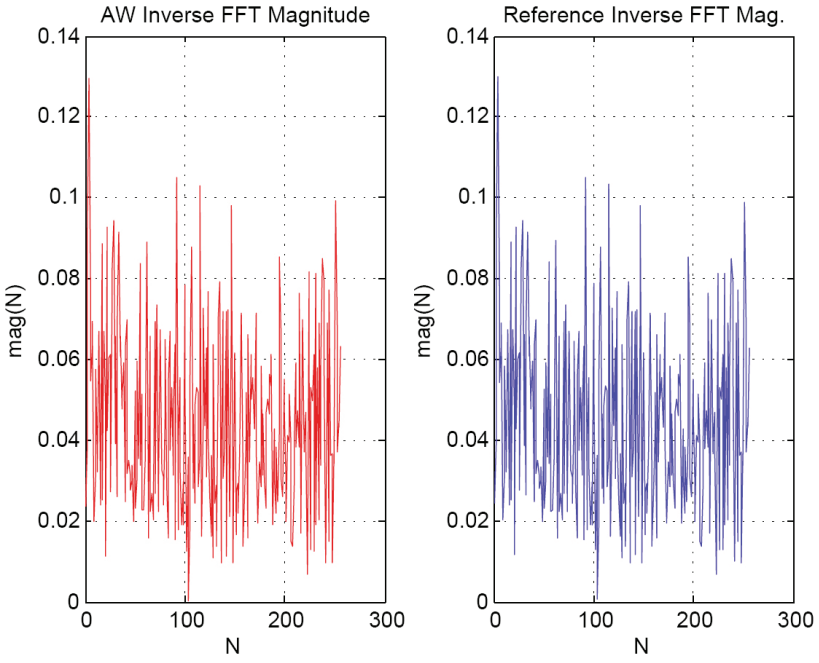


Fig. 7. Magnitude comparison between (a) fixed point model and (b) floating point model

The Angle comparison between fixed point model and floating point is presented in Fig. 8. As shown by Fig. 8 (a) and (b), the angle of signal in fixed point model of IFFT and the angle of signal in floating point model of the design overlap each other.

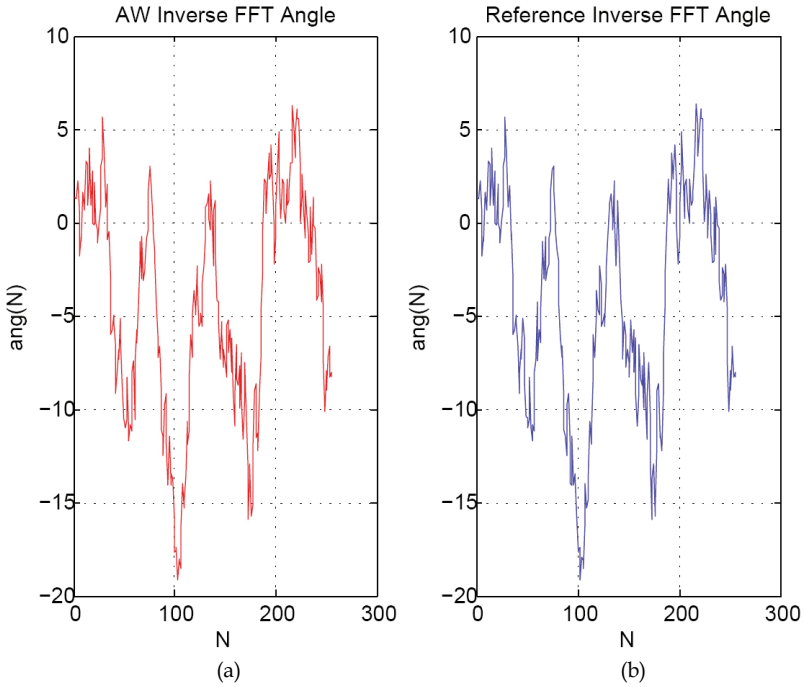


Fig. 8. Angel comparison between (a) fixed point model and (b) floating point model

The other form of verification can be performed using real and imaginary constellation graph. In order to verify designed IFFT, the AccelDSP tool generates real and imaginary constellation graphs and the verification can be done visually. Fig. 9 presents constellation based on IFFT with N=256 and Radix 2. In Fig. 9, the real part of fixed point model is shown by (a) which agree with (c) which is the real part in floating point model of IFFT design. The imaginary part of fixed point model is shown by (b) in Fig. 9 and it agree with (d) which is the imaginary part of floating point model of IFFT design.

As a result of discussed verifications, the error between fixed point model and floating point model is negative which is presented in Fig. 10. The other important parameters in designing hardware modules are hardware resource consumption and power consumption. These parameters can be estimated using Xilinx ISE tool. First the NGC Netlist file should be generated using System generator block in Matlab simulink and then through ISE the hardware consumption can be measured. In ISE GUI, from file folder, the saved project can be opened and then using Implement Top Module bottom, the hardware resource consumption table is generated which is presented in Table 1.

The main consideration is the percentages of DSP48 and IO Utilization. As shown in Table 1, the DSP48 and IO Utilization units of IFFT are used 6% and 16%, respectively.

The power consumed by the implemented DSI-SLM scheme is estimated by ISE XPower analyzer, Xilinx tool, after the place and route process. The processor consumes a total power of about 630 milliWatts and dynamic power of 10 milliWatts. Table 2 presents the details of power report. The ISE tool is also able to generate power consumption report.

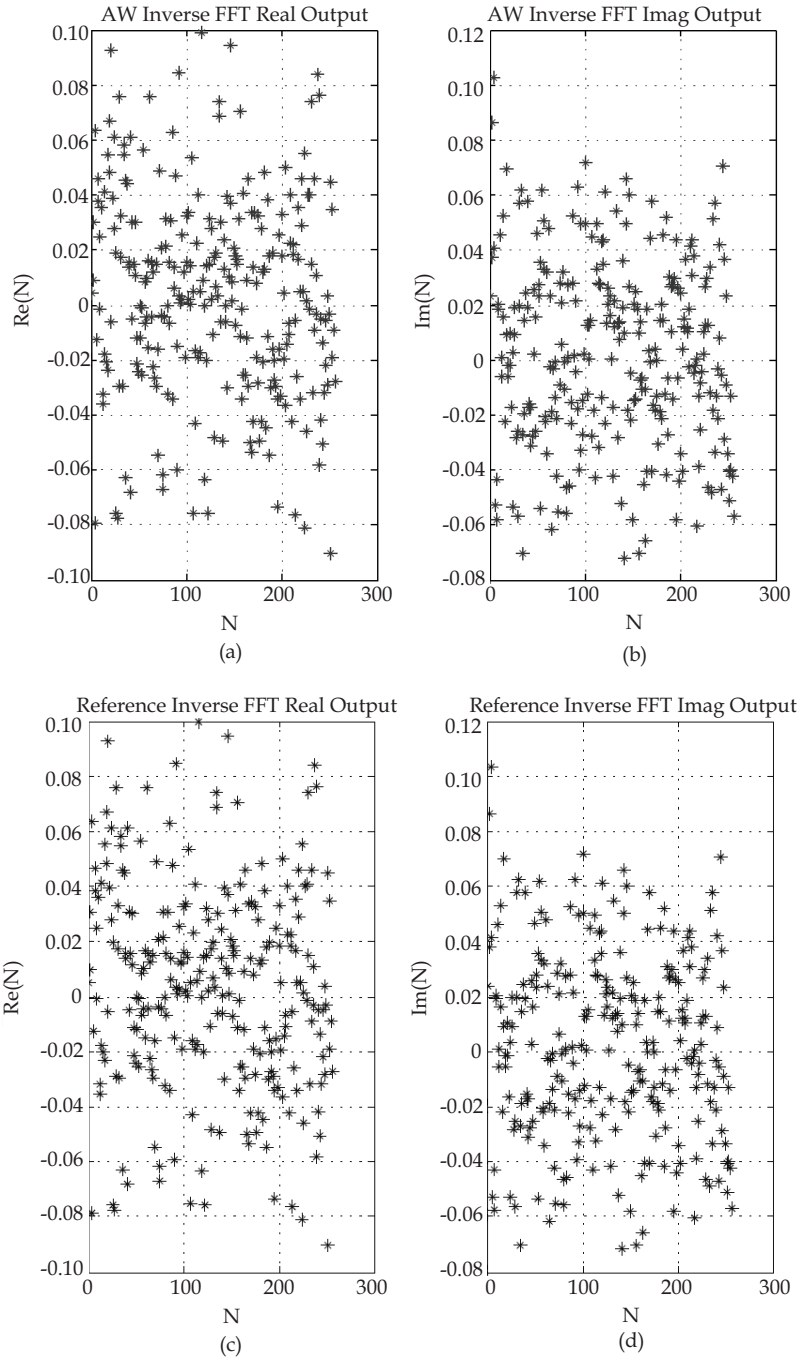


Fig. 9. Real and imaginary signal verification of IFFT

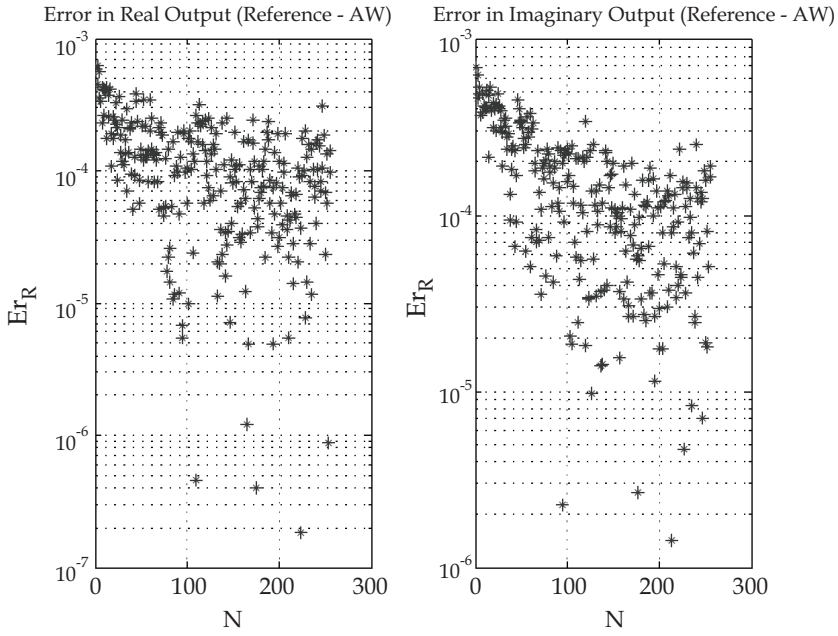


Fig. 10. Error study between fixed point model and floating point model

xc5vfx30t-1ff665	Resources Used	Percentages of Consumption
Slices	252	4%
DSP48 Slices	4	6%
Number of fully used LUT-FF pairs	455	56%
IO Utilization	58	16%

Table 1. Hardware resource consumption of IFFT block with Radix-2 and *N*=256

Name	Power [W]	Resources		
		Used	Total Available	Utilization [%]
Logic	0.00006	538	20480	2.6
Signals	0.00057	1103	---	---
DSP	0.00026	4	64	6.3
Total Quiescent Power	0.61977	N/A		
Total Dynamic Power	0.01030			
Total Power	0.63008			

Table 2. Power consumption report for IFFT with Radix-2 and N=256

As mentioned before, one of main consideration in designing OFDM systems is the computational complexity which can be defined by the number of real additions and multiplications that is required for hardware implementation of a system.

According to [Baxely et al., 2007] Each IFFT requires $N/2logN+N/2$ complex multiplication and $NlogN$ complex addition. A complex multiplication takes four real multiplications and two real additions. Total number of real addition required for an IFFT, A_{IFFT} is presented in Eq. (1) and total number of real multipications for an IFFT, M_{IFFT} is presented in Eq. (2). Hence the total number of multiplication and addition of one IFFT can be given by Eq. (3)

$$A_{IFFT} = 2(\frac{N}{2} log N + \frac{N}{2}) + 2N log N \tag{5}$$

$$M_{IFFT} = 4(\frac{N}{2} log N + \frac{N}{2}) \tag{6}$$

$$T_{IFFT} = A_{IFFT} + M_{IFFT} = 5N log N + 3N \tag{7}$$

where N is the number of subcarriers. When N=256, T_{IFFT}=3850. For N=512 and 1024, the value of T_{IFFT} is 8471 and 18484 respectively. It is obvious that by increasing the lenght of IFFT, the number of additions and multipication required for Implementing the IFFT is increased.

Design process of IFFT for higher number of subcarriers (N=512 and 1024) and Radix-4 is very similar to IFFT with Radix-2 and N=256. The Hardware resource Consumption and Power consumption for N=512 is presented in Table 3.

xc5vfx30t-1ff665	**Resources Used**	**Percentages of Consumption**
Slices	303	5%
DSP48 Slices	3	4%
Number of fully used LUT-FF pairs	554	64%
IO Utilization	58	16%

Table 3. Hardware resource consumption of IFFT block with Radix-2 and N=512

As shown in Table 3, the DSP48 and IO Utilization units of IFFT are used 4% and 16%, respectively.

According to ISE estimation, this IFFT processor consumes a total power of about 631 milliWatts and dynamic power of 11 milliWatts. Table 4 presents the details of this power report.

The Hardware Resource Consumption of IFFT processor with Radix-2 and N=1024 is presented in Table 5. It is shown that, the DSP48 and IO Utilization units of this IFFT block are used by 6% and 16%, respectively.

This IFFT processor consumes a total power of about 630 milliWatts and dynamic power of 10 milliWatts which are estimated by ISE tools. Table 6 presents the details of power report.

Name	Power [W]	Resources		
		Used	Total Available	Utilization [%]
Logic	0.00017	613	20480	3
Signals	0.00074	1198	---	---
DSP	0.00019	3	64	4.7
Total Quiescent Power	0.61993	N/A		
Total Dynamic Power	0.01197			
Total Power	0.63190			

Table 4. Power consumption report for IFFT with Radix-2 and N=512

xc5vfx30t-1ff665	Resources Used	Percentages of Consumption
Slices	330	6%
DSP48 Slices	4	6%
Number of fully used LUT-FF pairs	455	56%
IO Utilization	58	16%

Table 5. Hardware resource consumption of IFFT block with Radix-2 and N=1024

Name	Power [W]	Resources		
		Used	Total Available	Utilization [%]
Logic	0.00006	538	20480	2.6
Signals	0.00057	1103	---	---
DSP	0.00026	4	64	6.3
Total Quiescent Power	0.61977	N/A		
Total Dynamic Power	0.01030			
Total Power	0.63008			

Table 6. Power consumption report for IFFT with Radix-2 and N=1024

The Hardware Resource Consumption of IFFT processor with Radix-4 and N=256 is presented in Table 7. It is shown that, the DSP48 and IO Utilization units of this IFFT block are used by 18% and 16%, respectively.

xc5vfx30t-1ff665	Resources Used	Percentages of Consumption
Slices	696	13%
DSP48 Slices	12	18%
Number of fully used LUT-FF pairs	1064	49%
IO Utilization	58	16%

Table 7. Hardware resource consumption of IFFT block with Radix-4 and N=256

When comparing Table 7 and Table 1, it can be observed that the IO Utilization has no changes. However the consumption of DSP48 slices is increased by about 12%.

The IFFT processor with Radix-4 and *N*=256 consumes a total power of 0.65099 Watt and dynamic power of 0.02939 Watt which are estimated by ISE tools. Table 8 presents the details of power report. By Comparing Table 8 with Table 2, it is seen that power consumption of Radix-4 is increased compared to Radix-2 by about 0.02 Watt.

Name	**Power [W]**	**Resources**		
		Used	**Total Available**	**Utilization [%]**
Logic	0.00098	1364	20480	6.7
Signals	0.00322	2775	---	---
DSP	0.00078	12	64	18.8
Total Quiescent Power	0.62160	N/A		
Total Dynamic Power	0.02939			
Total Power	0.65099			

Table 8. Power consumption report for IFFT with Radix-4 and N=256

xc5vfx30t-1ff665	**Resources Used**	**Percentages of Consumption**
Slices	629	12%
DSP48 Slices	12	18%
Number of fully used LUT-FF pairs	1001	51%
IO Utilization	58	16%

Table 9. Hardware resource consumption of IFFT block with Radix-4 and *N*=1024

The Hardware resource consumption of IFFT processor with Radix-4 and *N*=1024 is presented in Table 9. It is shown that, the DSP48 and IO Utilization units of this IFFT block are used by 18% and 16%, respectively.

Name	**Power [W]**	**Resources**		
		Used	**Total Available**	**Utilization [%]**
Logic	0.00080	1364	20480	6.7
Signals	0.00317	2775	---	---
DSP	0.00078	12	64	18.8
Total Quiescent Power	0.62125	N/A		
Total Dynamic Power	0.02573			
Total Power	0.64698			

Table 10. Power consumption report for IFFT with Radix-4 and N=1024

The IFFT processor with Radix-4 and N=1024 consumes a total power of 0.64698 Watt and dynamic power of 0.02573 Watt which are estimated by ISE tools. Table 10 presents the details of power report.

5. Recent application of IFFT processor

Simple structure of an OFDM symbol consist of 4 sinusoids is shown in Fig. 11. The OFDM signal is created by the sum of multiple sinusoidal signals. Due to the constructive interference, as shown in Fig. 3 high peaks will be structured and as a result of destructive interference, the average power might be as low as zero. Hence, the ratio between peak and average will be high (Higashinaka et al., 2009).

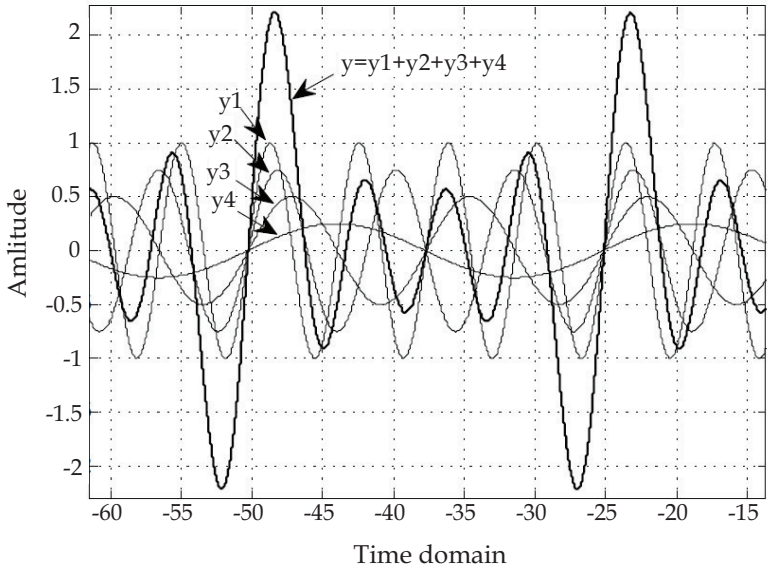


Fig. 11. Sample of OFDM signal behavior

High Peak to Average Power Ratio (PAPR) is a major design challenge in OFDM systems (Krongold et al., 2003; Bauml et al., 1996; Wei et al., 2006).

The reason is that when OFDM signal with high PAPR is introduced to amplification stage, Power Amplifier (PA) which is usually peak power limited, is forced to operate in the non-linear region (Nieto, 2005). This will cause two impacts, out-of-band distortion or spreading the spectrum that can be measured by Adjacent Channel Power Ratio (ACPR) metric and in-band distortion, which can be measured by Error Vector Magnitude (EVM) metric.

There are some PAs with a wide dynamic linear region (Class AB), however they are generally expensive, consume more power, and less efficient (Cooper, 2008; Varahram et al., 2009; Sharma et al., 2010). Hence, in order to have high-efficiency OFDM signal and extended battery life, the PAPR must be reduced and the linearity of PA should be maximized.

The PAPR is calculated as the ratio of the maximum power and the average power of signal and can be defined by:

$$PAPR(A) = \frac{max\left[\left|s(t)\right|^2\right]}{E\left[\left|s(t)\right|^2\right]} \tag{8}$$

where $s(t)$ is the N carriers OFDM envelope presented as below (Ochiai, 2003):

$$s(t) = \sum_{n=0}^{N-1} A_n e^{j\omega_0 nt} \tag{9}$$

where $A = (A_0, A_1, \ldots, A_{(N-1)})$ is a modulated data sequence of length N in the time interval $(0,T)$, where A_i is a symbol from a signal constellation and T is the OFDM symbol duration.

where, $\omega_0 = 2\pi/T$ and $j = \sqrt{-1}$.

Basically the performance of a PAPR reduction is measured using Complementary Cumulative Distribution Function (CCDF) graph. It denotes the probability that the PAPR of a data symbol exceeds a predefined threshold as expressed by (Han et al., 2005; Heo et al., 2009):

$$\begin{aligned} &probability(PAPR \rangle z) = 1 - probability(PAPR \le z) \\ &= 1 - F(z)^N = 1 - (1 - exp(-z))^N \\ &F(z) = 1 - exp(z) \end{aligned} \tag{10}$$

where N is the number of subcarriers and z is the threshold. Basically, this probability function is used as a graph to determine the ability of an algorithm in reducing the PAPR of the OFDM signal and the PAPR is usually compared to unmodified OFDM signal at 0.01% CCDF which is shown by 10^{-4} CCDF in horizontal vector of graphs. A typical OFDM signal without any PAPR reduction technique has about 8dB to 13dB PAPR at 10^{-4} CCDF (Raab et al., 2011). Therefore, when a PAPR reduction technique is applied to the OFDM system, it is expected to reduce the 13dB PAPR to some lower value. According to the IEEE standard (IEEE STD 802.16e™-2005), the reduction should be at least 3dB.

Several techniques have been developed to reduce PAPR of the OFDM signal. There are two main categories for these techniques, distortion based methods (which means that applying these methods result in out-of-band distortion) and distortion less methods (there is no out-of-band distortion). First category includes Clipping (May et al., 1998), Windowing (Van et al., 1998), Envelope Scaling (Foomooljareon et al., 2002), Random Phase Updating (Nikookar et al., 2002), Peak Reduction Carrier (Tan et al., 2003), Companding (Hao et al., 2006; Hao et al., 2010; Cao et al., 2007; Chang et al, 2010; Hao et al., 2008; Kim et al., 2008) and other modified version of these methods.

Clipping is a simple technique for PAPR reduction, where in the transmitter, the signal is clipped to a desired level and the phase information remains unchanged. The clipping method applies distortion to the system; therefore normally clipping technique is integrated with filtering method in expense of additional IFFT and FFT blocks which increase the

complexity of the system. In windowing technique a large signal peak is multiplied with a certain frame. Envelope scaling method is an algorithm to reduce PAPR by scaling the input envelope for some subcarriers before they are sent to IFFT. In the random phase updating algorithm, some random phases are generated and assigned for each carrier. The process of updating is continued till the peak value of the OFDM signal is below the threshold. The peak reduction carrier involves the use of a higher order modulation scheme to represent a lower order modulation symbol (Vijayarangan et al., 2009). The Companding technique is used to compress and expand the OFDM signal in order to reduce PAPR. The speech processing is the main application of companding method as it has less frequent peaks problem.

The second category of PAPR reduction methods is named distortionless techniques. These methods have significant PAPR performance without causing nonlinear distortion. However, they typically incur large computational complexities and sometimes side information transmission. Moreover, these methods usually require receiver side modifications that may be incompatible to existing communication systems. Such approaches include Coding (Jones et al., 1994; Kwon et al., 2009), Partial Transmit Sequence (PTS) (Muller et al., 1997; Gao et al., 2009; Chen et al., 2010; Kang et al., 1999), Selected Mapping (SLM) (Bauml et al., 1996,), Dummy Signal Insertion (DSI) (Ryu et al., 2004; Qian et al., 2005), Tone Injection and Tone Reservation (Tellado, 2000), Interleaving (Jayalath et al., 2000), Active Constellation Extension (ACE) (Krongold et al., 2003).

Most of the recent researches are concentrating on modified SLM and PTS methods (Wang et al. 2011; Ghassemi et al. 2010; Naeiny et al., 2011; Kim et al., 2006; Jeon et al., 2011; Hong et al., 2010). According to the review, most of the modified methods reduce PAPR at the expense of complexity in the transmitter or degrading the spectrum efficiency of the system. It should be noted that improving the performance of SLM based techniques requires high number of IFFT processors which leads to high complexity. The PTS based methods also have drawback of complexity from another aspect. The improvement of these methods requires extra number of additions and multiplication to be implemented for finding optimum value which leads to high complexity. Hence, there is good scope to design a new method to overcome previous drawbacks and enhance the PAPR performance.

Here one of recently proposed methods for reducing PAPR of OFDM signal is presented (Mohammady et al., 2011). This method is named Optimum Phase Sequence insertion with Dummy Sequence Insertion (OPS-DSI). As shown in block diagram of OPS-DSI scheme in Fig. 13, there are two loops in OPS-DSI algorithm. If the PAPR is not less than the threshold, *Loop_a* with specific number of iterations is performed and the PAPR will be compared. If the PAPR is less than the threshold, the signal will be transmitted regardless of the second loop, otherwise the second loop *Loop_b* with predefined number of iterations is executed and the PAPR is calculated similarly. When *Loop_a* is performed, a new random dummy is generated and inserted to the signal, howeverthe phase sequence is the same as last iteration. It should be noted that the number of iterations is specified based on the PAPR reduction requirement and data rate. The value of the PAPR threshold is also based on each standard in wireless broadband.

It should be noted that when *Loop_b* is running, *Loop_a* is repeated. It means that in *Loop_b,* new random phase sequence will be selected and multiplied to the signal and then a new random dummy is inserted to the signal. When the threshold condition is passed, the signal

will be transmitted, however if the iterations for both loops are performed and still the PAPR is not less than the threshold, the signal with minimum PAPR among them will be transmitted.

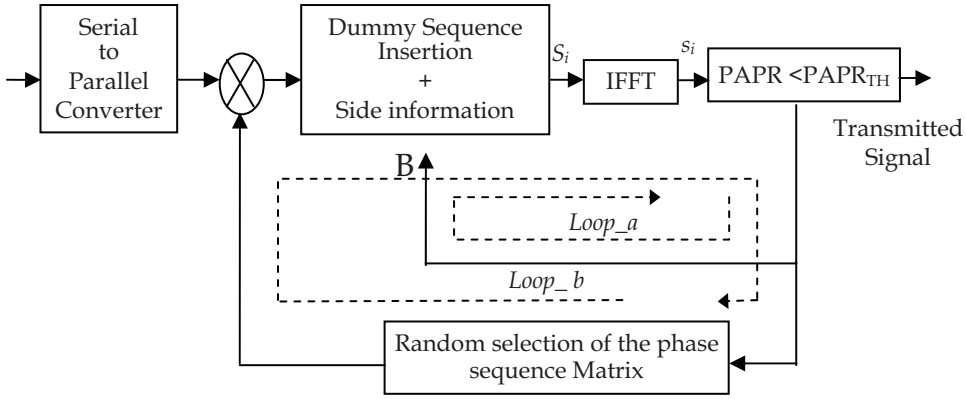


Fig. 12. Block diagram of the OPS-DSI scheme, transmitter

The CCDF result of a typical OFDM system with OPS-DSI scheme is compared with C-SLM and DSI methods. As shown in Fig. 13 (a), the PAPR of original OFDM signal is about 11.8dB at 10^{-4} CCDF or 0.01% CCDF. When DSI method is applied to this system, the PAPR is reduced to about 9.9dB shown by Fig. 13 (b) which means that the PAPR performance is enhanced by about 1.9dB compared to original OFDM signal.

When the C-SLM method with 8 IFFTs (number of candidate signals, M=8) is applied to the OFDM signal, the PAPR of 8.5dB is achieved which is shown by Fig. 13 (c). In this case, the PAPR is enhanced by about 3.4dB.

It is shown by Fig. 13 (d) that when OPS-DSI scheme is applied to the OFDM signal, the PAPR of about 7.7dB is achieved. In other words, the PAPR is enhanced by about 4.2dB compared to original signal. The PAPR performance of implemented OPS-DSI scheme is shown by Fig. 13 (e). The implemented system shows slightly degraded PAPR performance which is due to the Hardware input bit resolution. The ISE tol is able to generate total data path delay for OPS-DSI design which is 10.937 ns. This delay is within the accepted range according to Shannon-Hartly theorem (Hartley, 1928).

While comparing this result with recent works (Jeon et al., 2011; Naeiny et al., 2011; Hong et al., 2010; Wang et al., 2011; Kim et al., 2006), the face that 4.2dB reduction is achieved with only one IFFT and lowest complexity makes OPS-DSI method a very attractive method suitable for FPGA implementation.

In some literature papers, the PAPR performance is studied using time domain symbols. Fig. 14 presents 1024 samples of output signal with and without PAPR. Blue color samples are the output signal without PAPR reduction and the red color samples are the output signal when PAPR reduction is applied. It can be observed that the OFDM signal peaks are suppressed. However the reduction seems to be insignificant. The reason is that OPS-DSI scheme is a probabilistic method and the reduction is based on signal modification, therefore, time domain graph is not an accurate study tool for this case.

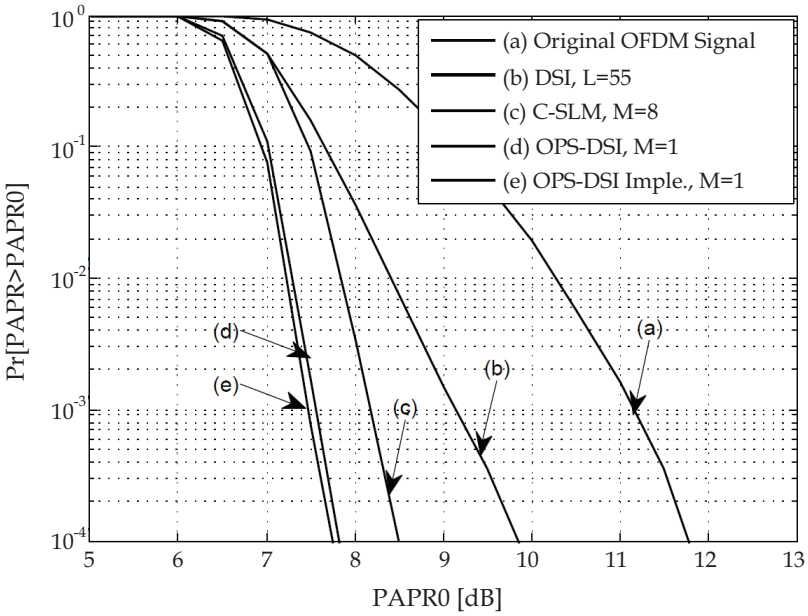


Fig. 13. Comparison of PAPR performance, (a) Original OFDM signal, (b) DSI method, (c) C-SLM method, (d) OPS-DSI method in simulation, (e) OPS-DSI in Implementation

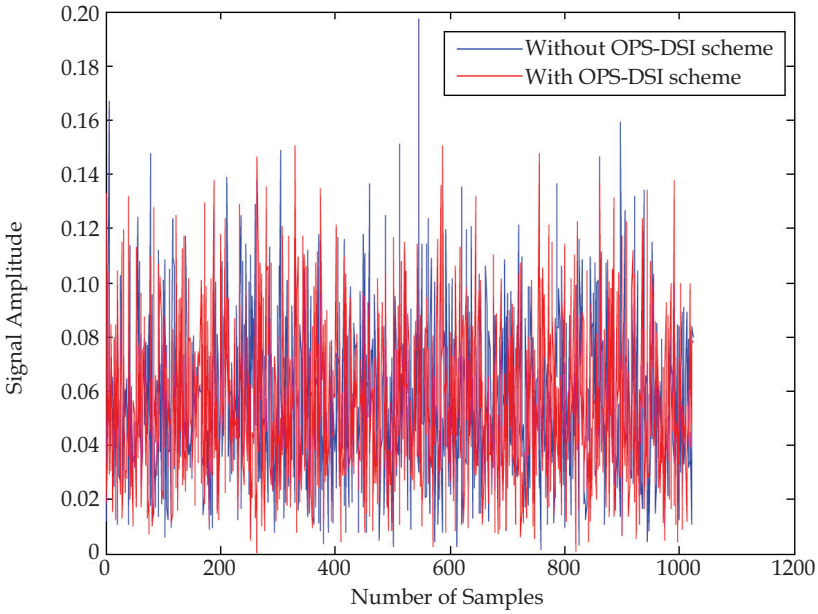


Fig. 14. OFDM signal in time domain, (Blue) without PAPR reduction technique, (Red) with PAPR reduction technique

6. Conclusion

In this chapter the OFDM transmition system is studied. The main component of IFFT processor is introduced. Hardware implementation of this block is performed and results are compared with the simulation. A very iportant application of IFFT in PAPR reduction scheme of OPS-DSI is reviewed. Different type of algorithms for IFFT are tested and Hardware resource consumption and power consumption are estimated using ISE tools. The complexity of implementing one IFFT block in the FPGA is mathematicaly computed.

7. References

Bauml R. W., Fischer R. F. H., Huber J. B. (1996). Reducing the peak-to-average power ratio of multicarrier modulation by selected mapping. *Electronics Letters, 32*(22): 2056-2057.

Baxley, R. J., & Zhou, G. T. (2007). Comparing selected mapping and partial transmit sequence for PAR reduction. *Broadcasting, IEEE Transactions on, 53*(4): 797-803.

Cao R., Jiang, T., Qin, J. (2007). Study on companding transforms for reduction in PAPR of OFDM signals. *Tien Tzu Hsueh Pao/Acta Electronica Sinica, 35*(6): 1099-1101.

Chang, P., Jeng, S., Chen, J. (2010). Utilizing a novel root companding transform technique to reduce PAPR in OFDM systems. *International Journal of Communication Systems,* 23: 447–461.

Chen, J. (2010). Application of quantum-inspired evolutionary algorithm to reduce PAPR of an OFDM signal using partial transmit sequences technique. *IEEE Transactions on Broadcasting,* 56(1): 110-113.

Cooper, S., (2008). Digital Radio Techniques for Energy Efficient OFDM Base stations, Axis Network Technology [White Paper], Retrieved March 2009 from http://www.axisnt.com/downloads/DigitalRadioWP.pdf

Foomooljareon P., Fernando W.A.C. *Input sequence envelope scaling in PAPR reduction of OFDM*. Proceedings of the 5th International Symposium on Wireless Personal Multimedia Communications, Honolulu, 27-30 Oct. Hawaii, 2002.

Gao J., Wang J., Wang B. (2009). Peak-to-average power ratio reduction based on cyclic iteration partial transmit sequence. *3rd International Symposium on Intelligent Information Technology Application, IITA 2009,* 2: 161-164.

Ghassemi A., Gulliver, T. (2010). PAPR reduction of OFDM using PTS and error-correcting code subblocking. *IEEE Transactions on Wireless Communications,* 9(3): 980-989.

Han S. H., Lee J. H. (2005). An overview of peak-to-average power ratio reduction techniques for multicarrier transmission. *IEEE Wireless Communications, 12*(2): 56-65.

Hartley R.V.L. (1928). Transmission of Information. *Bell System Technical Journal*.

Heo S., Joo H., No J., Lim D., Shin D. *Analysis of PAPR reduction performance of SLM schemes with correlated phase vectors*. Proceedings of the IEEE International Symposium on Information Theory (ISIT 2009), Seoul, June 28-July 3 2009, Korea, 2009.

Hao M., Liaw C. (2008). A companding technique for PAPR reduction of OFDM systems. *IEICE Transactions on Communications*, E91-B(3): 935-938.

Hao, M., Lai, C. (2010). Precoding for PAPR reduction of OFDM signals with minimum error probability. *IEEE Transactions on Broadcasting,* 56(1): 120-128.

Hao, M., Liaw, C. *A companding technique for PAPR reduction of OFDM systems*. Proceedings of International Symposium on Intelligent Signal Processing and Communications (ISPACS'06), Yonago, 12-15 Dec. Japan, 2006.

Higashinaka M., Fukui N., Kubo H. (2009). On peak to average power ratio of generalized frequency division multiple access. *IEICE Electronics Express*. 6 (13): 943-948.

Hong, E., Har, D. (2010). Peak-to-average power ratio reduction in OFDM systems using all-pass filters. *IEEE Transactions on Broadcasting,* 56(1): 114-119.

Jayalath A., Tellambura C. (2000). Reducing the peak-to-average power ratio of orthogonal frequencydivision multiplexing signal through bit or symbol interleaving. *Electronics Letters. 36*(13): 1161-1163.

Jeon H. B., No J. S., Shin D. J. (2011). A Low-Complexity SLM Scheme Using Additive Mapping Sequences for PAPR Reduction of OFDM Signals. *IEEE Transactions on Broadcasting*, PP(99): 0-1.

Jones A., Wilkinson T., Barton S. (1994). Block coding scheme for reduction of peak to mean envelope power ratio of multicarrier transmission schemes. *Electronics Letters, 30*(25): 2098-2099.

Kang S. G., Kim J. G., Joo E. K. (1999). A novel subblock partition scheme for partial transmit sequence OFDM. *IEEE Transactions on Broadcasting*. 45(3): 333-338.

Kim S. W., Chung J. K., Ryu H. G., PAPR Reduction of the OFDM Signal by the SLM-based WHT and DSI Method, IEEE Region 10 Conference (TENCON-2006), Hong Kong, 14-17 Nov. China, 2006.

Krongold B. S., Jones D. L. (2003). PAR reduction in OFDM via active constellation extension. *IEEE Transactions on Broadcasting. 49*(3): 258-268.

Kwon J. W., Park S. K., Kim Y. *Peak-to-average power ratio reduction by the partial shift sequence method for space-frequency block coded OFDM systems*, Proceedings of 2009 IEEE International Conference on Network Infrastructure and Digital Content (IEEE IC-NIDC2009), Beijing 6-8 Nov. China, 2009.

Kwon U., Kim D., Im G. (2009). Amplitude clipping and iterative reconstruction of MIMO-OFDM signals with optimum equalization. *IEEE Transactions on Wireless Communication., 8*(1): 268-277.

May, T., Rohling, H., *Reducing the peak-to-average power ratio in OFDM radio transmission systems*, Proceedings of 48th IEEE Vehicular Technology Conference, VTC 1998, Ottawa, Canada, May 18-21 1998.

Muller S. H., Huber J. B. (1997). OFDM with reduced peak-to-average power ratio by optimum combination of partial transmit sequences. *Electronics Letters, 33*(5): 368-369.

Naeiny M. F., Marvasti F. (2011). Selected mapping algorithm for PAPR reduction of space-frequency coded OFDM systems without side information. *IEEE Transactions on Vehicular Technology,* 60(3): 1211-1216.

Nieto J.W. *An investigation of coded OFDM and CEOFDM waveforms utilizing different modulation schemes on HF channels*, Proceedings of the 6th International Symposium

on Communication Systems, Networks and Digital Signal Processing (CNSDSP), Graz, July 23-25 Austria, 2008.

Nikookar H., Lidsheim K. S. (2002). Random phase updating algorithm for OFDM transmission with low PAPR. *IEEE Transactions on Broadcasting, 48*(2): 123-128.

Ochiai H. (2003). Performance analysis of peak power and band-limited OFDM system with linear scaling. *IEEE Transactions on Wireless Communications, 2*(5): 1055-1065.

Pratt T. G., Jones N., Smee L., Torrey M. (2006). OFDM link performance with companding for PAPR reduction in the presence of non-linear amplification. *IEEE Transactions on Broadcasting, 52*(2): 261-267.

Qian H., Xiao Ch., Chen N., Zhou G.T., *Dynamic selected mapping for OFDM*. Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05). Philadelphia, PA, 18-23 March USA. 2005.

Qian, H. (2005). *Power Efficiency improvements for wireless transmissions. Power Efficiency Improvements for Wireless Transmissions,* Doctoral dissertation, School of Electrical and Computer Engineering, Georgia Institute of Technology, USA.

Ryu H. G., Lee J. E., Park J. S. (2004). Dummy sequence insertion (DSI) for PAPR reduction in the OFDM communication system. *IEEE Transactions on Consumer Electronics. 50*(1): 89-94.

Sharma P.K., Basu A. *Performance Analysis of Peak-to-Average Power Ratio Reduction Techniques for Wireless Communication Using OFDM Signals*, Proceedings of the International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom), 16-17 Oct. India, 2010.

Tan C.E., Wassell I.J. *Data bearing peak reduction carriers for OFDM systems. Information*, Proceedings of the 2003 Joint Conference of the Fourth International Conference on Communications and Signal Processing the Fourth Pacific Rim Conference on Multimedia, December 15-18, Singapore, 2003.

Tellado J. (2000). *Multicarrier modulation with low PAR. Applications to DSL and wireless* Kluwer Academic Publication. Dordrecht, Netherlands, 2000.

Tellado, J. (2000). *Peak to average power reductionfor multicarrier modulation.* Doctoral dissertation, Stanford University, USA.

Van N. R., De W. A. *Reducing the peak-to-average power ratio of OFDM*. Proceedings of 48th IEEE Vehicular Technology Conference (VTC 1998), Ottawa, May 18-21 Canada, 1998.

Varahram P., Mohammady S., Hamidon M.N., Sidek R.M., Khatun S. (2009). Digital predistortion technique for compensating memory effects of power amplifiers in wideband applications. *Journal of Electrical Engineering.* 60(3): 129-135.

Vijayarangan V., Sukanesh D. (2009). An overview of techniques for reducing Peak to Average Power Ratio and its selection criteria for orthogonal frequency division multiplexing radio systems. *Journal of theoretical and applied information technology.* 5(1): 25-36.

Wang L., Liu J. (2011). PAPR Reduction of OFDM Signals by PTS with Grouping and Recursive Phase Weighting Methods, *IEEE Transactions on Broadcasting.* 57(2): 299-306.

Wang S. H., Sie J. Ch., Li Ch. P., Chen Y. F. (2011) A Low-Complexity PAPR Reduction Scheme for OFDMA Uplink Systems. *IEEE Transactions on Wireless Communications*, 10(4): 1242-1251.

Wei G., Hu L., Yu H. (2006). PAR reduction for OFDM and DMT signals based on distribution transform. *Chinese Journal of Electronics. 15*(2): 282-286.

7

Fault Diagnosis of Induction Motors Based on FFT

Castelli Marcelo, Juan Pablo Fossatti and José Ignacio Terra
Universidad de Montevideo
Uruguay

1. Introduction

The FFT (Fast Fourier Transform) can be used for on-line failure detection of asynchronous motors. In this work a methodology is described for the most likely to happen faults in induction motors: broken rotor bars, bearing damage, short circuits and eccentricity.

In industrialized countries, induction motors are responsible for the 40% to 50% of energy consumption. The electric motors in most cases are responsible for the proper functioning of the productive system. In this line, the corrective maintenance of equipment is very expensive since it involves unscheduled downtime and damage to the production process caused by equipment failures.

Nowadays, there are many published techniques and allowed commercial tools for the induction motors failure detection. Despite this, most industries still do not use detection and monitoring techniques of electrical machines.

Here a methodology for monitoring and diagnosis of induction motors is presented, which monitors the engine without removing it from the production line, being this methodology: reliable, easy to apply and low cost.

Next a brief study of the most likely faults to occur in induction motors has been made, as well as the existing methods for detecting them.

Then, a methodology based on the measurement of the stator current signal has been developed. This can show the frequency and magnitude of each failure happen to occur in this kind of engines.

After that, the methodology has been carried out to validate in the laboratory.

2. Failures in induction motors

Most failures in induction motors can be classified in two main groups: isolation failures and mechanical failures (Botha, 1997).

Isolation failures are commonly characterized by stator coils short-circuits, while mechanical faults are commonly associated to rotor or rotor related damage. The most important mechanical failures are: rotor broken bars and rings, bearings damage, irregular gaps (static and dynamics eccentricities), unbalances, refrigeration troubles, etc.

In general, faults in electrical machines are dominated by failures in bearings and stator coils. Focused on asynchronous motors with squirrel cage rotor failure statistics are the following (Fig. 1) (Thomson & Fenger, 2001):

- Bearings fault related: 41%
- Stator faults related: 37%
- Rotor faults related: 10:
- Other problems: 12%

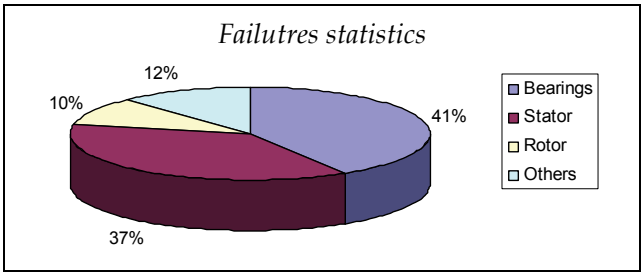


Fig. 1. Failures statistics in induction motors

The previous analysis of the most likely to happen failures in asynchronous motors showed the tree types that represent the 90% of the possible failures in those motors. Being the punctual failures:

- Bearing fault related
- Rotor broken bars and rings.
- Stator coils short-circuits
- Stator unbalance and eccentricities

2.1 MCSA (Motor Current Signature Analysis)

The main objective of the developed methodology is the motor slip determination using only the stator current. This parameter could be used for many applications, but this work is focused on fault detection via MCSA (Motor Current Signature Alalysis).

The on-line diagnosis via MCSA is used to detect several faults such as: stator and rotor failure, bearing damage and eccentricities in induction motors (Thomson, 1999) (Thomson & Gilmore, 2003). This method is based on the fact that an unbalanced machine when supplied with a three phase balanced voltage produces specific components in the stator current, whose magnitude and frequency depends on the asymmetry level and nature of the fault (Bellini et al, 2002). This method is based on the current signal spectrum decomposition, analyzed via the Fast Fourier Transform. The fault causes a harmonic component in the current at a characteristic frequency, visualized in the current spectrum. To determine these frequencies the motor slip should be known.

2.2 Eccentricity

There are two types of air gap eccentricity: the static air gap eccentricity and the dynamic air gap eccentricity. In the case of the static air gap eccentricity, the position of minimal radial

air-gap length is fixed in space. Thus for example, static eccentricity can be caused by the ovality of the core or by the incorrect positioning of the stator or rotor at the commissioning stage. In the case of the dynamic eccentricity the center of the rotor is not at the center of rotation and the position of the minimum air-gap rotates with the rotor. For example, dynamic eccentricity can be caused by a bent rotor shaft, wear of bearings, misalignment of bearings, mechanical resonances al critical speed and so on.

The frequency components in the stator current of an induction machine which are due to air gap eccentricity can be obtained as:

$$f_{ec} = f_1 \pm k.f_{rm} \tag{1}$$

Where,
f_{ec} = eccenticity frequency
f_1 = supply frequency
f_{rm} = rotor mechanical frequency
k = 1, 2, 3….

2.3 Slip estimation methodology

Here a slip estimation methodology using the frequency stator current spectrum is presented. The main objective is to identify the eccentricity associated component and therefore obtain the rotor speed.

First, the motor nominal speed must be known in order to determine the frequency range in which the eccentricity component can be found. Then, from (ec1) the fault associated frequency is calculated for nominal rotor speed.

$$f_{ecn} = f_1 \pm f_{rmn} \tag{2}$$

Where,
f_{ecn} = eccentricity frequency (at motor nominal speed)
f_1 = supply frequency
f_{rmn} = rotor nominal mechanical frequency

It is important to say that $k = \pm 1$ because the range in which the eccentricity frequency can be found is reduced to the maximum.

After the motor's nominal eccentricity frequency calculation, the motor current is determined, in order to specify if the eccentricity frequency (f_{ec}) is lower o higher than the nominal eccentricity frequency (f_{ecn}). For example, if we take $k = -1$ and the motor current is higher, then the eccentricity frequency is lower than the nominal eccentricity frequency. Table 1 shows the relationship between intensity, eccentricity frequency and rotor mechanical frequency and their nominal values. Where I is the measured current and I_N the nominal current.

Then, to bound the range of values where the eccentricity associated component can be found, reasonable limits for the motor speed have been established. For example, a motor with two pairs of poles and a measured current higher than the nominal, whose maximum rotor mechanical frequency is 25 Hz. Working with k=-1 it is possible to establish a minimum limit for the eccentricity frequency.

$$f_{ec}(\min) = f_1 - f_{rm}(\max) \tag{3}$$

$k = 1$		$k = -1$	
$I > I_N$	$I \le I_N$	$I > I_N$	$I \le I_N$
$f_{ec} < f_{ecn}$	$f_{ec} \ge f_{ecn}$	$f_{ec} > f_{ecn}$	$f_{ec} \le f_{ecn}$
$f_{rm} > f_{rmn}$	$f_{rm} \le f_{rmn}$	$f_{rm} < f_{rmn}$	$f_{rm} \ge f_{rmn}$

Table 1. Relationship between intensity, eccentricity frequency and rotor mechanical frequency

Then, the eccentricity frequency is bounded as follows:

$$f_{ec}(\min) \le f_{ec} \le f_{ecn} \tag{4}$$

If the nominal speed is 1435 rpm and the supply frequency is 50 Hz, the eccentricity frequency will be in a 1.08 Hz wide window.

$$25Hz. \le f_{ec} \le 26.08Hz. \tag{5}$$

If the stator current is higher than the nominal ($f_{rm} < f_{rmn}$), for setting a maximum limit for the eccentricity frequency a minimum rotor speed should be set. In this work the maximum slip limit was taken as 5 %, other slip values could be considered depending on the case.

$$f_{ecn} < f_{ec} \le f_{ec}(\max) \tag{6}$$

Where $f_{ec}(\max)$ calculation is based on considering a maximum slip of 5%. For example if we consider a motor with two pairs of poles with nominal speed of 1435 rpm and supply frequency of 50 Hz, the eccentricity frequency is bounded as follows:

$$26.08Hz < f_{ec} \le 26.25Hz \tag{7}$$

The eccentricity frequency can be found in a 0.17 Hz window, this allows an almost immediate identification. For the motor tested the sampling time was 8 seconds, thus a 0.125 Hz resolution. This makes possible an accurate identification of the eccentricity component and little motor speed variation.

Another important thing is that there is no mask effect, since the eccentricity frequency is far enough from the supply frequency associated component.

Once the eccentricity associated component is calculated, its associated frequency has to be determined. Being a_k the kth harmonic associated to the discrete Fourier Transform (DFT) and tm the sampling time, the eccentricity frequency is determined as follows:

$$f_{ec} = \frac{k \pm \frac{1}{1+\Psi}}{tm} \tag{8}$$

Where,

$$\Psi = \frac{|a_k|}{Max(|a_{k+1}|,|a_{k-1}|)} \tag{9}$$

Being a_k the main harmonic associated with the eccentricity frequency. It is important to say that ec8 is valid when there are no other components due to different causes next to the interest component, and that no temporary windows are used.

Finally, once the eccentricity associated frequency is found, it is possible to find from ec1 the rotor mechanical frequency and so, the slip.

2.4 Data acquisition model

There are several techniques that can be used for detecting faults in induction motors. The MCSA (Motor Current Signal Analysis) is a non-invasive, on-line monitoring technique for diagnosing problems in induction motors. This method is based on the spectral decomposition of the steady state stator current which can be acquired with simple measurement equipment and under normal operation of the machine. MCSA can diagnose problems such as broken rotor bars, shorted turns, bearing damage and air gap eccentricity. In the MCSA method the current frequency spectrum is obtained and analyzed aiming to find out specific components which can indicate an incipient fault in the machine. These frequencies are related to well-known machine faults. Therefore, after the processing of the stator current, it is possible to infer about the machine's condition.

A methodology with an accurate comprehension of the different influence of the variables is desired for the correct interpretation of the data acquired. In this work the frequency spectrum is obtained by the fast Fourier transform (FFT). For the case that the data acquisition is for a entire number of cycles of a certain component is easy to obtain its amplitude and frequency, this is not always the case. In consequence we will have in the frequency spectrum certain components that could mask others of interest. This is commonly known as leakage. Another thing to take into account is the fact that the motor's load condition is not always the same; this makes the fault signature characteristics different. The main objective of this methodology is to monitor these frequencies independently of the motor functioning and the data acquisition in order to determine the condition of the machine. To avoid the masking effect, the signal is multiplied by a function (window) reducing the discontinuity. In this opportunity we are not going to analyze the use of the different windows, but we focus in the acquisition of the signal's amplitude introduced by the failure.

2.5 Monitoring and diagnosis methodology

Here the fault frequency equations for the measured current signal deduced via the analysis of Fast Fourier Transform for each defect are presented.

2.5.1 Broken bars

For the case of broken rotor bars the equation is the following:

$$f_b = f_1(1 \pm 2.s) \tag{14}$$

Where,

f_b = broken bars frequency
f_1 = supply frequency
s = slip frequency

2.5.2 Short-circuits

For short-circuits in induction motors:

$$f_{st} = f_s.\left\{\frac{n}{p}.(1-s) \pm k\right\} \tag{15}$$

Where,
f_{st} = short-circuit frequency
f_s = supply frequency
n = 1, 2, 3…
p = pole pairs
s = slip frequency
k = 1, 3, 5…

2.5.3 Eccentricities

For the eccentricities, the equation is the following:

$$f_{ec} = f_s.\left[(r \pm n_d).\left(\frac{1-s}{p}\right) \pm n_{ws}\right] \tag{16}$$

Where,
f_{ec} = eccentricity frequency
f_s = supply frequency
r = number of slots
$n_d = +/-1$
p = pole pairs
s = slip frequency
n_{ws} = 1, 3, 5…

2.5.4 Bearing failures

Finally the equation for bearing failure:

$$\begin{aligned} f_0 &= 0.4 * n * f_{rm} \\ f_1 &= 0.6 * n * f_{rm} \end{aligned} \tag{17}$$

Where,
f_0 = lower frequency
f_1 = upper frequency
n = balls number
f_{rm} = rotor mechanical frequency

The equation 17 applies only when cyclic faults occur and only for bearings between 6 and 9 balls.

It is necessary to know the motor's simplified equivalent circuit to study. Based on the simplified equivalent circuit, and doing current and voltage measurement, it is possible to find the slip motor. It should be borne in mind that the slip found in this way is very sensitive to small variations of the parameters calculated before. Therefore the realization of an FFT of the current signal is necessary, as the harmonics of eccentricity of the motor are always present in this spectrum and this harmonics give us, from the applications of equations: 12 and 13, an accurate measure of the fault frequency, which form equation 18 provides a very precise measurement of slip frequency, which is used to identify all the failure frequencies.

$$f_{ec} = f_s \pm k \cdot f_{rm} \tag{18}$$

Where
f_{ec} = eccentricity frequency
f_s = supply frequency
k = 1, 2, 3…
f_{rm} = rotor mechanical frequency

From the current sampling and the slip frequency of the motor, it is possible to determine the frequencies for each failure study.

It is necessary to remark the fact that in the data acquisition, the FFT of the measured current was normalized as a function of the primary harmonic amplitude, in order to obtain fault amplitude independent form the motor load condition.

After determining the true amplitudes and frequencies of defect, it is necessary to study its evolution in time, for the selected motors.

2.6 Exact amplitude and frequency determination

Here the mathematical development in order to determine the exact amplitude and frequency of the failure.

First a brief description of the signal current reconstruction is shown. Then, a mathematical development is presented based on an accurate sampling of N periods of the signal current. After that the problems due to sampling of no-exact number of cycles are shown, and later the mathematical development that eliminates that problem.

$X[n]$	Discrete Fourier Transform
$x[n]$	Discrete function
N	Number of total samples
T	Time period
t	Time
$n = 0, \pm 1, \pm 2, ...$	
$k = 0, \pm 1, \pm 2, ...$	
ω_0	Fundamental frequency (for discrete and continuous)
a_k	Spectral coefficient (for discrete and continuous)
t_m	Sampling time
ω_s	Sampling frequency
ω_m	Minimum Sampling frequency

$\bar{x}$	Data vector
$\bar{X}$	Data vector ($\bar{X} = FFT(\bar{x})$)
$x(t)$	Signal as a function of time
j	Imaginary unit
Ψ	$\Psi = \frac{\lvert a_k \rvert}{\lvert a_{k\pm1} \rvert}$ relationship between the main harmonic and the one that follows in amplitude
y	Real value that indicates the number of cycles sampled above or below (depending on their sign) of the k cycles when evaluation the k-th harmonic
y'	Real value that indicates the number of cycles sampled above or below (depending on their sign) of the k cycles when evaluation the k-th harmonic, being the last the main harmonic.
T	Period of a function

2.7 Signal reconstruction

One of the most important issues to take into account for the proper failure detection is the signal interpretation. The signal is composed of sinusoidal continuous waves at different frequencies. These components will be in part caused by different defects that may exist and the characteristics of the load. Another aspect to consider is that the methodology should be versatile to adapt to different working conditions.

From the stator current sampling the different frequencies that and the wave amplitudes should be obtained. Then with these parameters the failure (the frequency) and its importance (amplitude) is obtained.

To reconstruct the original signal, among other things, the Fast Fourier Transform (FFT) will be used. The FFT is an algorithm that can efficiently calculate the discrete Fourier transform (DFT). This algorithm was created in 1965 by JW Cooley and JW Tuckey

The direct calculation of the DFT involves a number of operations proportional to N^2 (where N is the number of samples) while using the fast Fourier transform the number of operations is proportional to $NLog_2(N)$. This markedly reduces the calculating time of the discrete Fourier transformed. This last fact combined with the increased processing power of computers made the FFT become a fundamental tool in many disciplines. Communications, signal processing, sonar, radar, biomedical engineering, inventory analysis, metallurgy and applied mechanics are just some of the areas where the FFT is applied.

Note that to properly apply the fast Fourier transform is not necessary to have a thorough knowledge of the algorithm. Instead it is important to understand the discrete Fourier transform and therefore the Fourier transform.

2.8 Mathematical development

The following is an introduction to Fourier series representation of discrete periodic signals and discrete Fourier transform (DFT). Similar to continuous functions, the fundamental period N is defined and the fundamental frequency ω_0 , such that: $x[n] = x[n+N]$

where N is the smallest integer that fulfills the above relationship.

$$\omega_0 = \frac{2\pi}{N} \tag{19}$$

Then $\theta_k[n]$, that we define as:

$$\theta_k[n] = e^{j\omega_0 kn} = e^{j\frac{2\pi}{N}kn} \tag{20}$$

With k = p, p+1,…, (N-1) + p (with p integer) or $k = \langle N \rangle$

then $\theta_k[n] = \theta_{k+rN}[n]$ (with r integer).

As follows $x[n]$ as a combination of complex exponentials:

$$x[n] = \sum_{k=\langle N \rangle} a_k \theta_k[n] = \sum_{k=\langle N \rangle} a_k e^{j\omega_0 kn} = \sum_{k=\langle N \rangle} a_k e^{j\frac{2\pi}{N}kn} \tag{21}$$

Where a_k can be expressed:

$$a_k = \frac{1}{N} \sum_{n=\langle N \rangle} x[n] e^{-j\omega_0 kn} \tag{22}$$

With $a_k = a_{k+N}$ and $a_k = \overline{a_{-k}}$ (only when $x[n]$ is real for all *n*).

In a similar way the Discrete Fourier Transform (DFT)is defined as:

$$x[k] = \sum_{n=\langle N \rangle} x[n] e^{-j\frac{2\pi}{N}nk} = \sum_{n=\langle N \rangle} x[n] e^{-j\omega_0 nk} \tag{23}$$

The inverse Discrete Fourier Transform (IDFT) can be expressed as follows:

$$x[n] = \frac{1}{N} \sum_{k=\langle N \rangle} X[k] e^{j\frac{2\pi}{N}nk} = \frac{1}{N} \sum_{k=\langle N \rangle} X[k] e^{j\omega_0 nk} \tag{24}$$

With the algorithm FTTW, Matlab obtains the DFT from a data vector composed:

$$\bar{x} = [x[0], x[1], ..., x[N-1]] \tag{25}$$

Is the same as: $\bar{x}(n) = x[n-1]$

When doing the FFT($\bar{x}$) an $\overline{X}$ vector is obtained given by the following expression:

$$X(k) = \sum_{i=1}^{N} x(i) \omega_n^{(i-1)(k-1)} \tag{26}$$

With $\omega_n = e^{-\frac{2\pi}{N}j} = e^{-\omega_0 j}$

The k-th value of the $\overline{X}$ vector corresponds to (k-1)t harmonic:

$$\overline{X}(k) = X[k-1] = a_{k-1}N \tag{27}$$

Or,

$$\overline{X}(k) = [a_0, a_1, ..., a_{N-1}]N \tag{28}$$

In order to know the reach of the DFT and make a correct interpretation some examples are shown:

Figure 2 shows the sampling of a sinusoidal signal of amplitude 1.

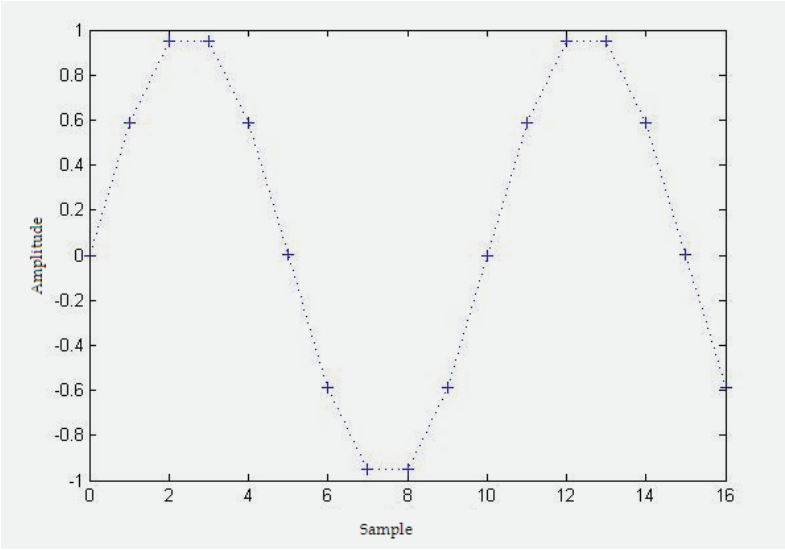


Fig. 2. Sinusoidal signal

If N=10 and the samples were taken during one period:

$$\overline{x} = [\sin(\frac{2\pi}{10}0), \sin(\frac{2\pi}{10}1), \sin(\frac{2\pi}{10}2), ..., \sin(\frac{2\pi}{10}9)] \tag{29}$$

It is possible to express $x[n]$ as follows:

$$x[n] = \frac{1}{2j}e^{j\frac{2\pi}{10}n} - \frac{1}{2j}e^{-j\frac{2\pi}{10}n} \tag{30}$$

Which is the same as:

$$x[n] = \frac{1}{2j}e^{j\frac{2\pi}{10}n} - \frac{1}{2j}e^{j\frac{2\pi}{10}9n} \tag{31}$$

By doing the FFT $\bar{x}$ an $\bar{X}$ vector is obtained, where:

$$\bar{X}(2) = X[1] = Na_1 = \frac{N}{2j} \tag{32}$$

$$\bar{X}(9) = X[9] = Na_9 = -\frac{N}{2j} \tag{33}$$

For the other positions: $\bar{X}(i) = 0$.

Supposing 10 samples of the same function with a phase shift: $\theta = \frac{2\pi}{10}3$.

The following data vector $\bar{x}$ is obtained

$$\bar{x} = [\sin(\frac{2\pi}{10}3), \sin(\frac{2\pi}{10}4), \sin(\frac{2\pi}{10}5), ..., \sin(\frac{2\pi}{10}12)] \tag{34}$$

$$x[0] = \sin(\frac{2\pi}{10}3)\,,\; x[1] = \sin(\frac{2\pi}{10}4)\,, \ldots,\; x[9] = \sin(\frac{2\pi}{10}12)$$

Then, it is possible to express $x[n]$ as follows:

$$x[n] = \frac{1}{2j}e^{j(\frac{2\pi}{10}n + \frac{2\pi}{10}3)} - \frac{1}{2j}e^{-j(\frac{2\pi}{10}n + \frac{2\pi}{10}3)} \tag{35}$$

Then:

$$\bar{X}(2) = X[1] = \frac{N}{2j}e^{j\frac{2\pi}{10}3} \tag{36}$$

$$\bar{X}(10) = X[9] = -\frac{N}{2j}e^{-j\frac{2\pi}{10}3} \tag{37}$$

To generalize, the DFT is analyzed for N samples of a sinusoidal function period with β amplitude and $\theta = \frac{2\pi}{N}d$ phase shift, with $d = 0, \pm 1, \pm 2...$ or $x[n] = \beta \sin(\frac{2\pi}{N}n + \frac{2\pi}{N}d)$. The following vector is obtained $\bar{X}$:

$$\bar{X}(2) = X[1] = \frac{\beta N}{2j}e^{j\frac{2\pi}{N}d} \tag{38}$$

$$\bar{X}(N) = X[N-1] = -\frac{\beta N}{2j}e^{-j\frac{2\pi}{N}d} \tag{39}$$

The wave amplitude is given by:

$$|X(2)|\frac{2}{N}=|X[1]|\frac{2}{N}=\beta \tag{40}$$

Taking N samples during two periods.

$$\begin{aligned} x[n] &= \beta\sin(2\frac{2\pi}{N}n+2\frac{2\pi}{N}d) \\ &= \frac{\beta N}{j2}e^{j\frac{2\pi}{N}2d}e^{j\frac{2\pi}{N}2n}-\frac{\beta N}{j2}e^{-j\frac{2\pi}{N}2d}e^{-j\frac{2\pi}{N}2n} \\ &= \frac{\beta N}{j2}e^{j\omega_0 2d}e^{j\omega_0 2n}-\frac{\beta N}{j2}e^{-j\omega_0 2d}e^{-j\omega_0 2n} \end{aligned} \tag{41}$$

For the generic case where k periods are sampled of a sinusoidal function of β amplitude and phase shift $\theta=\frac{2\pi}{N}d$, it is possible to express $x[n]$ as follows:

$$x[n]=\frac{\beta N}{j2}e^{j\omega_0 kd}e^{j\omega_0 kn}-\frac{\beta N}{j2}e^{-j\omega_0 kd}e^{-j\omega_0 kn} \tag{42}$$

So:

$$a_k=\frac{\beta N}{j2}e^{j\omega_0 kd} \text{ y } a_{N-k}=a_{-k}=-\frac{\beta N}{j2}e^{-j\omega_0 kd}$$

Therefore, if a vector is generated $\bar{x}$ = [$x[0],x[1],...x[N-1]$] then the FFT $\bar{X}=FFT(\bar{x})$ is as follows:

$$\bar{X}(k+1)=X[k]=a_k \tag{43}$$

$$\bar{X}(N-k+1)=X[n-k]=a_{N-k}=a_{-k} \tag{44}$$

For the rest of the values: $\bar{X}(i)=0$

If N samples of k periods for any sinusoidal function would be easy to find the original signal amplitude as shown by the following equation:

$$\beta=|\bar{X}(k+1)|\frac{2}{N}=|\bar{X}[n]|\frac{2}{N} \tag{45}$$

Being tm the sampling time, assuming complete cycles, then it is possible to deduce the exact frequency (f) and the amplitude.

$$f=\frac{k}{tm} \tag{46}$$

3. Laboratory tests results

In this section, laboratory tests results for: broken rotor bars, short-circuit, eccentricities and bearing failure are presented.

3.1 Broken rotor bars

The broken rotor bar simulation was achieved by perforation in the bars with a drill (Figure 3).

Fig. 3. Rotor perforations

First the tests were carried out with the healthy motor. Then the rotor was perforated in order to simulate the broken rotor bars failure (Fig. 4).

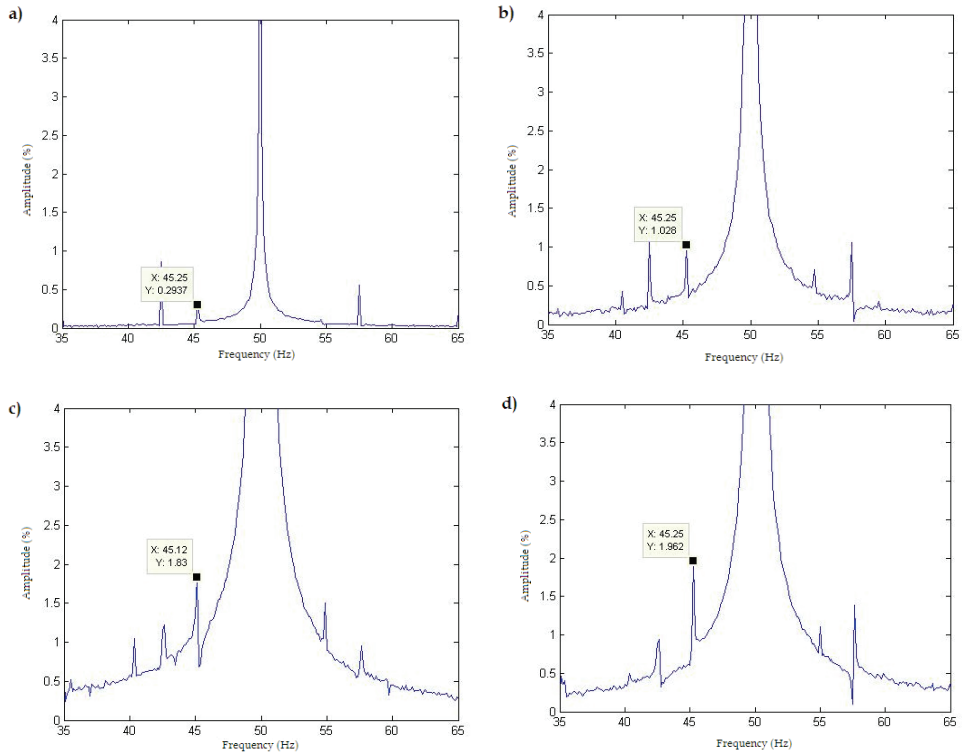


Fig. 4. a) Healthy motor; b) One broken bar; c) Two broken bars; d) Three broken bars

For all the test the amplitude was normalized to the main harmonic and the test were carried out with a constant charge (1429 rpm).

The frecuencies where should appear the failure components:

Revolutions: 1429 rpm

$$s = \frac{1500 - 1429}{1500} = \ 0.0473$$

$$fc = f \cdot (1 \pm 2.s)$$

$$fc^{+} = 54.7300 \text{ HZ}$$

$$fc^{-} = \ 45.2700 \text{ HZ}$$

For all of the cases the presence of the harmonics on that frequency can be verified. Also it is important to say that there is a clear relationship between the number of broken bars and the relative amplitude of the harmonics (Table 2) (Figure 5). It should be noted that in rotor broken bars failure, cancelation occurs if they have a phase shift of 180 electrical degrees between two broken bars.

Frequency (Hz)	Amplitude (%)	Number of broken bars
45.25	0,294	0
45.25	1,028	1
45.25	1,830	2
45.25	1,962	3

Table 2. Relative Amplitude for the different number of broken bars tested

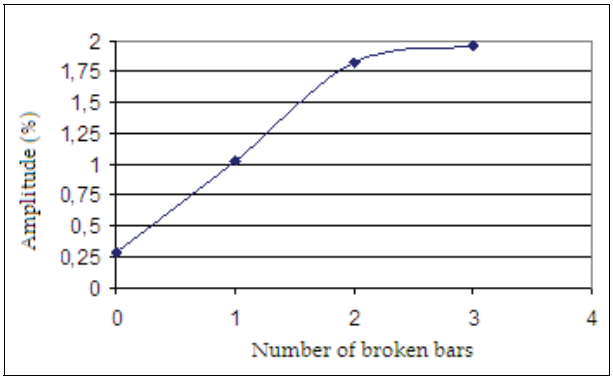


Fig. 5. Failure evolution (Amplitude vs. Number of broken bars)

As seen in Fig. 6 there is no appreciable difference between with one broken bar and five broken bars. This is due to the fact that there is electrical symmetry between 2 pairs of them. This reflects a limitation of the proposed methodology in terms of diagnostic capability of damage if they occur symmetrically.

This methodology not only detects the presence of a defect but also makes it possible to establish a trend. It is important for determining whether the engine should continue to operate or if necessary out of service (the latter is essential). On the other hand, is that the presence of symmetrical defects could be a drawback to detect broken bars. Finally, it is possible to use this type of industrial-level diagnosis for the detection of broken bars in induction motors.

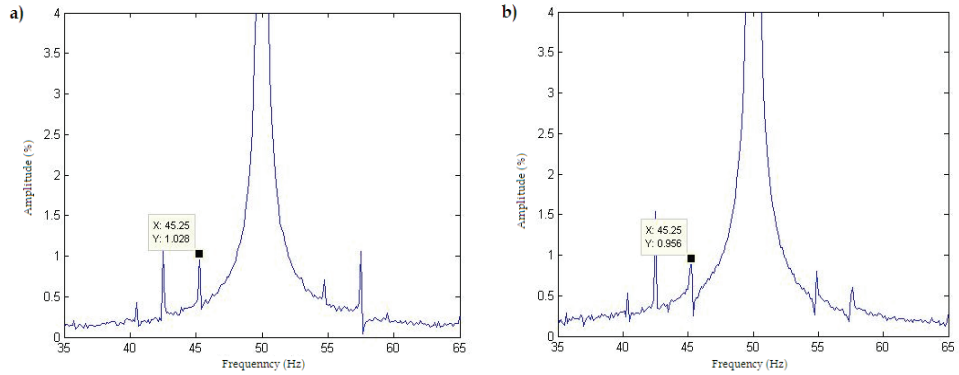


Fig. 6. a) One broken bar; b) Five broken bars

It must be said that cyclical loads can produce frequency responses similar to broken bar failures.

3.2 Short-circuit

Of all the types of short-circuit that can occur in the engine (Fig. 7), the ones that occur at the same stage are more difficult to detect (this is due to the fact that the protections are not activated). This is why the work focuses on this type of failure.

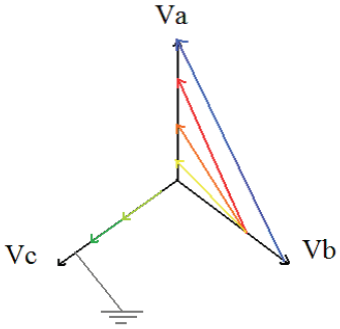


Fig. 7. Different types of short-circuit. In blue, orange, yellow and red the short-circuits are between two phases of different magnitudes. In gray a ground fault is represented and in green two shorts of the same phase of different magnitudes (these defects are the ones that will be simulated)

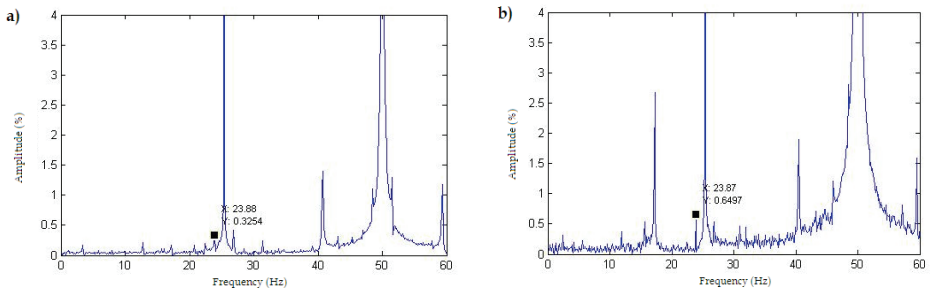
The potential difference was measured between the different points in order to determine the short-circuit magnitude. The result is shown in Table 3 which expresses the potential difference and the percentage that this represents. Fig. 8 shows how each cable was connected and then smoothed through a hole to simulate a short circuit.

Fig. 8. Induced short-circuit

	1	2	3	4	5	6	7
1							
2			0.51	1.58	0.64	5.81	1.93
3		1.95%		1.15	0.11	6.34	1.39
4		6.05%	4.41%		1.02	7.51	0.22
5		2.45%	0.42%	3.91%		6.47	1.27
6		22.26%	24.29%	28.77%	24.79%		7.75
7		7.40%	5.33%	.84%	4.87%	29.69%	

Table 3. Short-circuit magnitude

The tests were carried out for short-circuits of: 0% (healthy motor), 1.02%, 1.39% and 1.95%% (Fig. 9) at 1477rpm (s=0.015). For these work conditions the characteristically frequency for short-circuit 23.85 Hz (with n=3 y k=1).



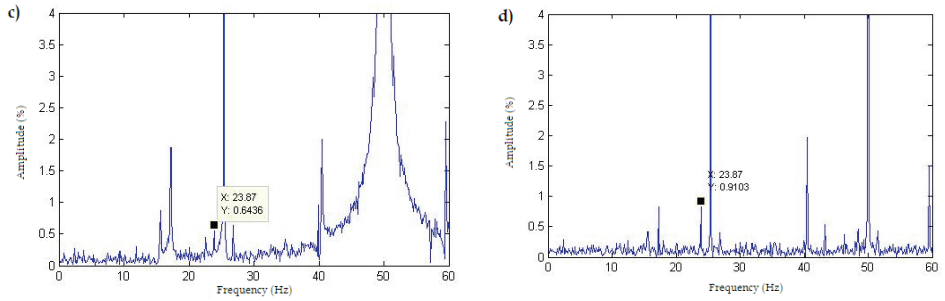


Fig. 9. a) Healthy motor; b) Short-circuit of 1.02 %; c) Short-circuit of 1.39 %; d) Short-circuit of 1.95 %

Frequency (Hz)	Amplitude (%)	Short-circuit (%)
23.84	0.80	0
23.84	1.34	1.02
23.85	1.47	1.39
23.85	1.89	1.95

Table 4. Amplitude for the different short-circuit magnitude

In table 4 the different short-circuits tested. The presence of a harmonic at a frequency of 23.84Hz coincides with the previous slip s measurement (23.85 Hz). Then, the short-cut magnitude (at 25.36 Hz) as a function of the defect amplitude (Fig. 10).

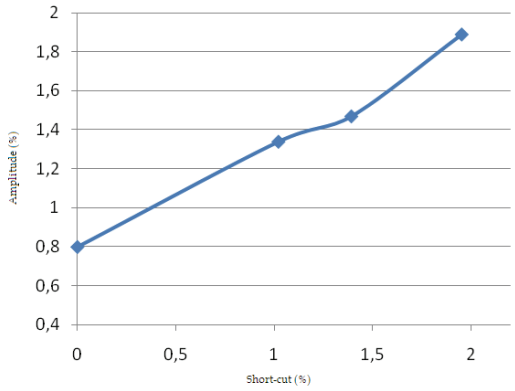


Fig. 10. Amplitude vs. short-cut magnitude

It can be concluded that there is a strong tendency as the magnitude of the defect increases above 1.0%, (Fig. 10). It is possible to predict the existence of a short circuit that due to its magnitude has not occurred to activate the protection, but would be able to leave the engine inoperative.

In high voltage motors it is very short the time between a short circuit within a phase and a short circuit between phases (seconds or minutes). This makes this methodology difficult to implement.

3.3 Bearing failure

First the tests were carried out with the healthy motor. Then the bearings were manipulated with a punch in order to simulate the bearing damage, one at the time (Fig. 11). The sampling was carried out at 5kHz during 8 seconds. In table 5 the rotor speed for each test, where test 1 is with the healthy motor, 2, 3 and 4 correspond to 1, 2 and 4 bearing damage.

Test	Speed (rpm)	Rotor frequency (Hz)
1	1482	24.70
2	1482	24.70
3	1482	24.70
4	1483	24.72

Table 5. Rotor speed and failure frequency for each test

It is possible to obtain the related failure frequency for bearing damage. For a rotor speed of 1482 rpm, the failure frequency is: 69.15 Hz and 69.20 Hz for a speed of 1483 rpm.

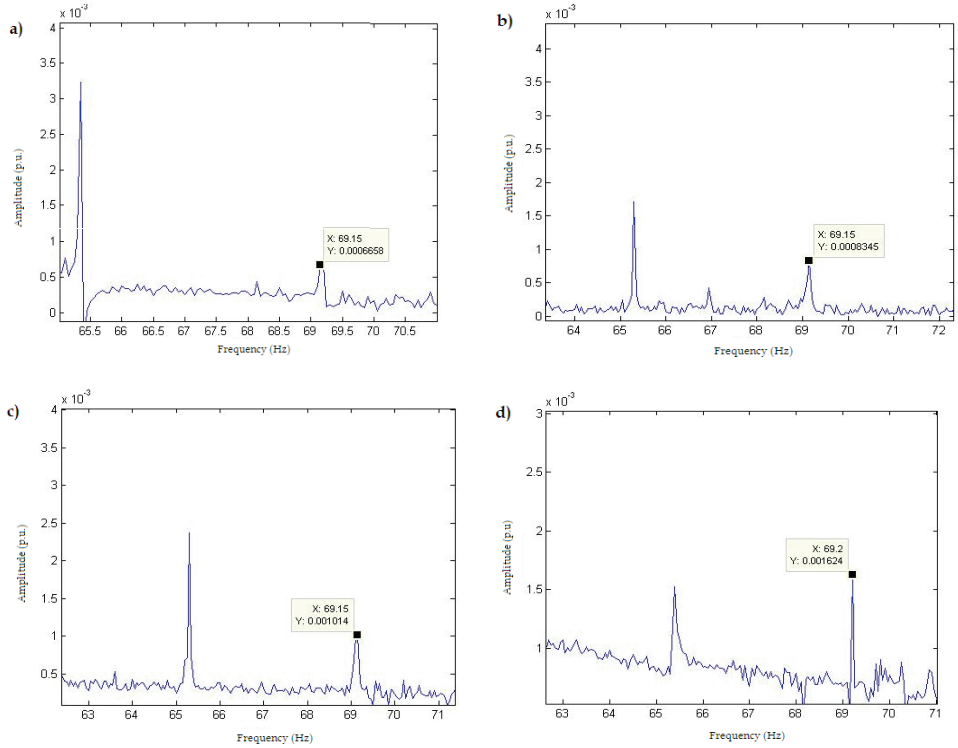


Fig. 11. a) Healthy motor; b) One bearing damage; c) Two bearings damage; d) Four bearings damage

The failure amplitude increases with the number of damaged bearings (Table 6) (Fig. 12). In Fig. 13 the damage simulated in the studied bearing.

Test	Failure Amplitude (%)
1	0.105
2	0.110
3	0.160
4	0.185

Table 6. Failure amplitude for each test

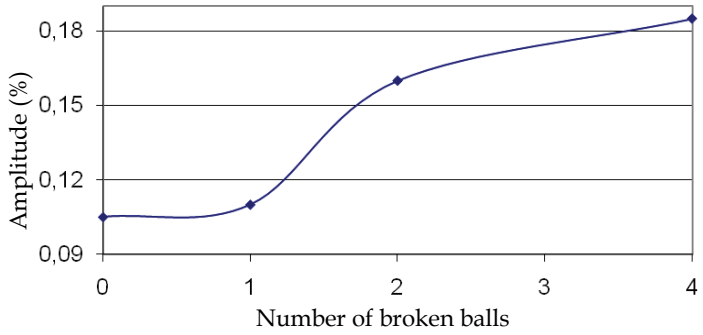


Fig. 12. Amplitude vs. number of damaged bearings

Fig. 13. Bearing damage

It is possible to detect bearing damage with the diagnostic methodology presented here. It is important to mention that none of the defects introduced into the engine in order to determine the course of the defect and the severity of the fault, have prevented the proper operation under the same load condition. Taking this into account it can be concluded that for this particular case, it is possible to detect the damage in time for scheduling preventive maintenance stop.

3.4 Eccentricities

In order to measure the eccentricity the center position of the rotor from a fixed reference was measured. Then the relative displacement of the rotor with respect to the baseline, as a percentage of nominal air gap of the machine was calculated. In Fig. 14 the eccentricity induced failure, in Fig. 15 the eccentricity generation.

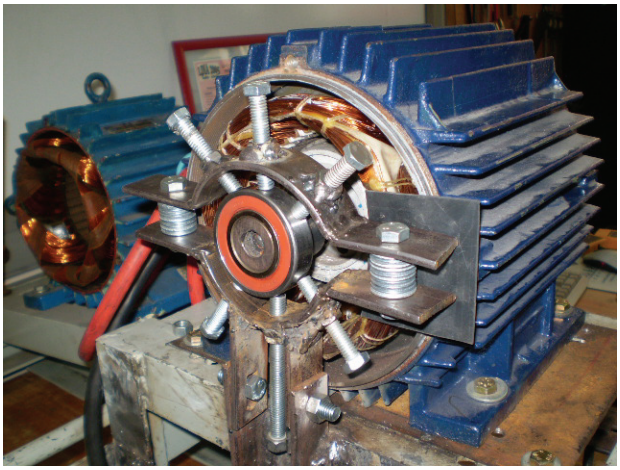


Fig. 14. Eccentricity induced failure

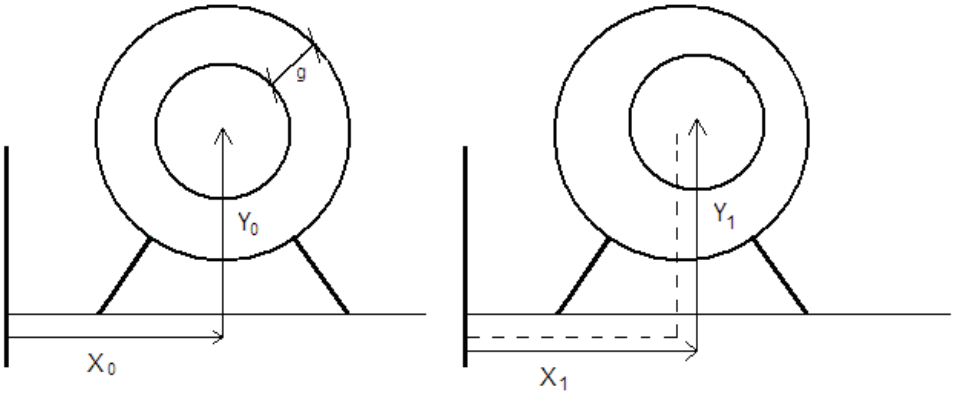


Fig. 15. Eccentricity generation

From the different measurements the eccentricity was calculated as a percentage using the following expression:

$$e(\%) = \frac{\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}}{g} \tag{47}$$

In table 7 the results for each test.

Test	Eccentricity (%)	speed (rpm)	Frequency (Hz)
1	0	1393	26.79
2	44,5	1400	26.67
3	87.3	1361	27.31

Table 7. Eccentricity results

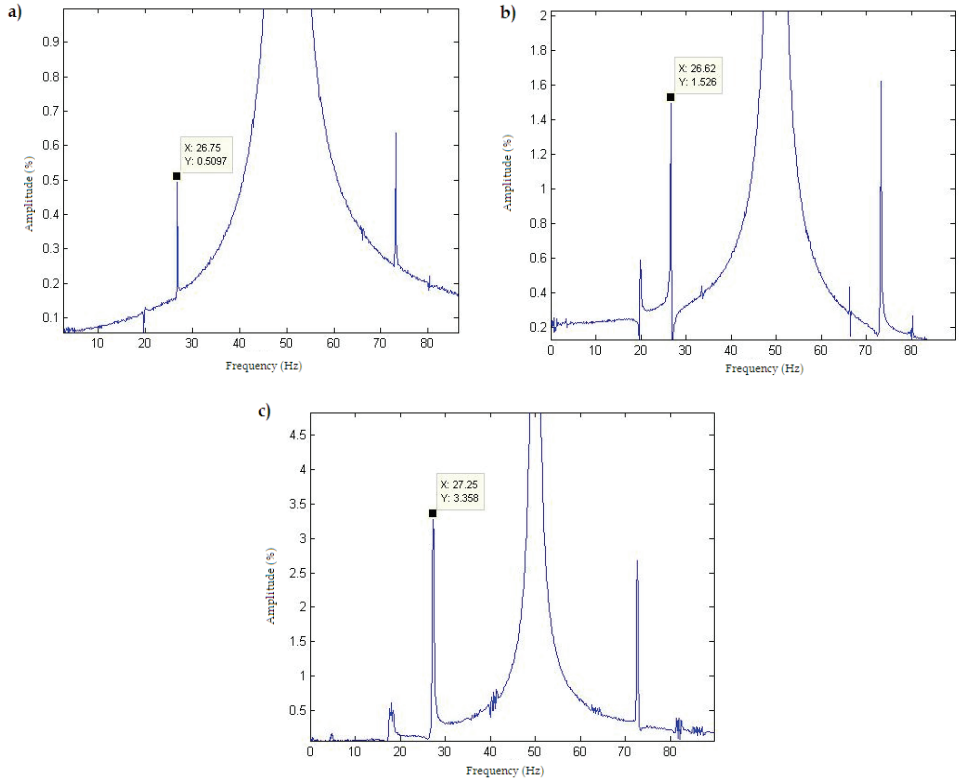


Fig. 16. a) Healthy motor; b) e=44.5%; c) e=87.3%

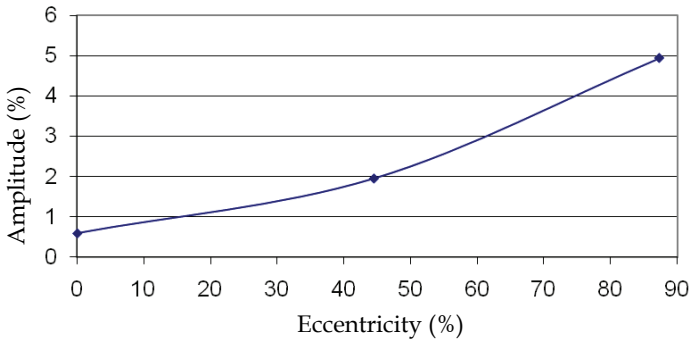


Fig. 17. Failure amplitude vs. eccentricity

There is a clear relationship between the failure amplitude and the amount of eccentricity (Table 8) (Fig. 17). In Fig. 18 the induced eccentricity failure.

The methodology proposed in this work is capable of detecting eccentricity failures and its evolution as a function of the defect amplitude. It should be noted however, that the tests

performed with the prototype built for this purpose, develop an intrinsic eccentricity defect. Thus, although the proposed methodology has been tested for the detection of this defect, at the industrial level should be monitored by studying the evolution of the defect in several tests and not from a standard that can be built from trials in the test.

Eccentricity (%)	Amplitude (%)
0	0.59
44,5	1.96
87,3	4.94

Table 8. Failure amplitude

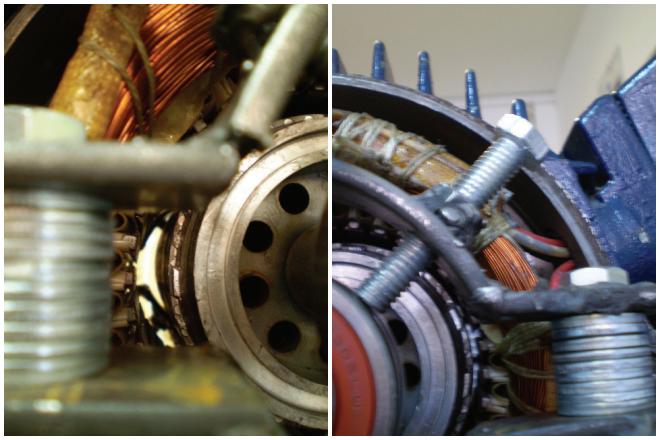


Fig. 18. Eccentricity failure

4. Industrial tests results

In this section, the results of industrial tests are presented. In table 9 the main characteristics of the motor tested (Fig. 19).

Fig. 19. Tested motor

POWER	250Hp
Number of pole pairs	3
Nominal Voltage	380V

Table 9. Main motor characteristics

The known information for the tested motor is poor. First the stator current was measured (Fig. 20). Then the normalized FFT of the sampled signal was carried out (Fig. 21).

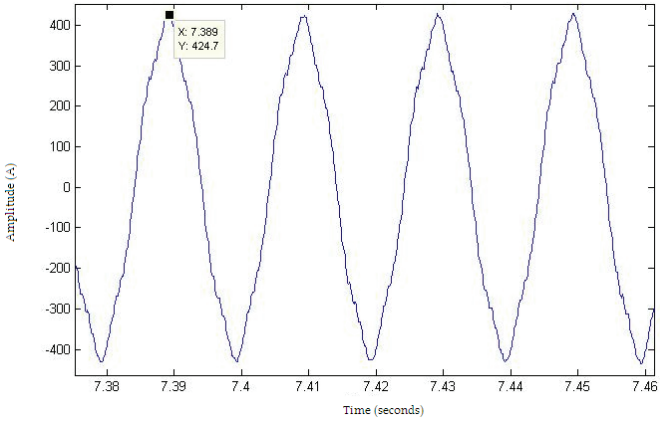


Fig. 20. Stator current

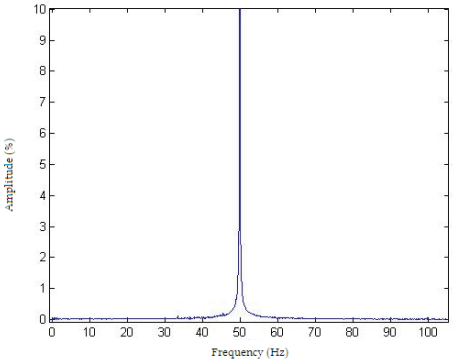


Fig. 21. Normalized FFT of the sampled signal

A nominal rotation speed near to 1000 rpm was assumed (synchronous motor speed). Knowing the nominal power of the engine, it can be deduced from the current consumption during the test that speed will be very close to that of synchronism.

From this hypothesis, we took a speed window from 970 rpm to 997 rpm to estimate the eccentricity frequency. For these speed values the eccentricity associated frequency corresponds to 33.83 Hz and 33.38 Hz. In Fig. 22 is possible to distinguish a peak on the 33.4 Hz.

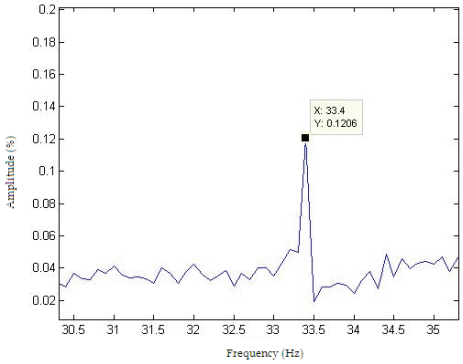


Fig. 22. f_{exc} determination

By calculating the exact associated eccentricity frequency from a_k and a_{k+1} harmonics, it is possible to determine that the engine speed is n_r = 994rpm, which is consistent with the previously known facts of the engine tested. In Figures 23, 24 25 y 26 the failure associated frequency is shown for each of the tested failure (broken bars, short-circuit, bearing damage and eccentricity).

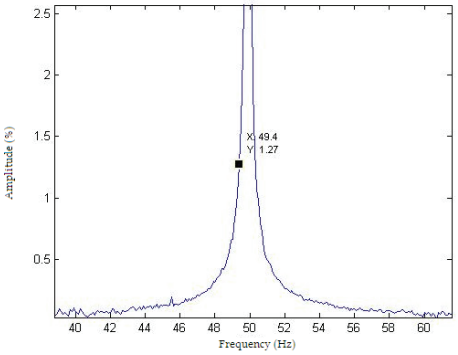


Fig. 23. Broken bars associated frequency

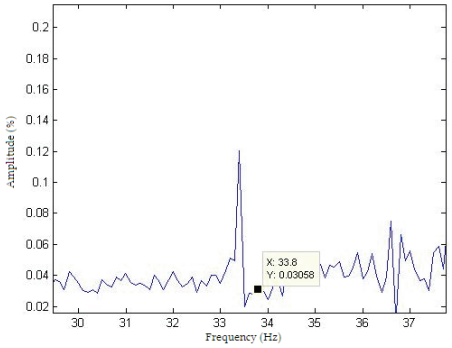


Fig. 24. Short-circuit associated frequency

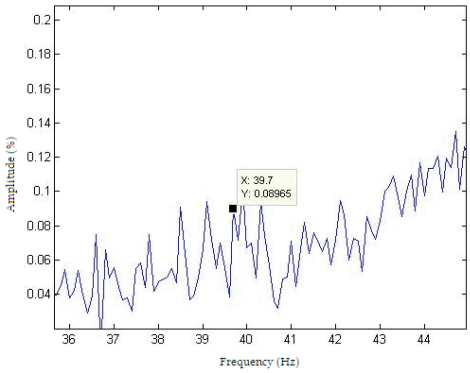


Fig. 25. Bearing damage associated frequency

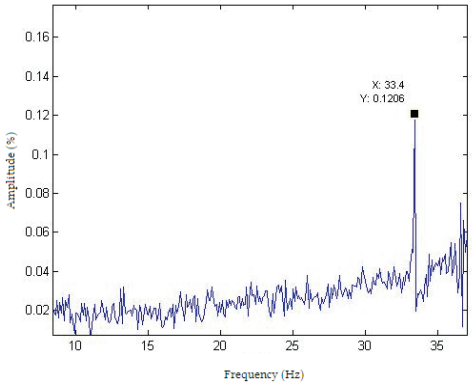


Fig. 26. Eccentricity associated frequency

From the figures presented above, it is possible to infer that the condition of the engine tested for defects studied is good. The only fault is raised above the standard is for broken bars, but this behavior is expected due to the proximity of the engine speed with its synchronous speed.

5. Summary and conclusions

The main objective of this work has been to determine a monitoring and diagnosis methodology for asynchronous motors which can be applied at the industrial level.

The proposed system is able to ascertain the exact value of the harmonic in study regardless of the sampling time.

This methodology gives encouraging results. It is possible from this technique to determine the exact extent of the defect and the associated frequency. In this way, the study of the growth tendencies of failures is easier. Also a practical slip estimation method for these motors has been introduced. This method requires minimum sensorization and the no need to remove the motor while doing the tests.

Laboratory tests results for broken rotor bars, short-circuit, eccentricities and bearing faults are shown. The method was validated for the failures studied. The results show that it is possible by this method to detect the failures in an incipient stage. Finally the results of industrial tests show the accuracy of the methodology.

6. References

Bellini, A. Filippetti, F. Franceschini, G. Tassoni, C. & Kliman, G. B. (2002); On-field experience with online diagnosis of large induction motors cage failures using MCSA, IEEE Trans. on Ind Applications, Vol. 38, Nº 4, pp. 1045-1053

Botha, M. M (1997). Electrical Machines Failure, Causes and Cures, Electrical Machines and Drives. 8th annual conference of IEEE, Nº 444, pp. 114-117, 1-3 de September of 1997

Thomson, W.T (1999). A Review of On-Line Condition Monitoring Techniques for Three-Phase Squirrel-Cage Induction Motors – Past, Present and Future, The Robert Gordon University, Schoolhill, Aberdeen, Scotland

Thomson, W.T. & Fenger, M. (2001). Current Signature Analysis to Detect Induction Motor Faults, IEEE Ind. Applications Magazine.

Thomson, W.T. & Gilmore, R. J. (2003). Motor Current Signature Analysis to Detect Faults in Induction Motor Drives- Fundamentals, Data Interpretation, and Industrial Case Histories. Proccedings of 32rd Turbomachinery Symposium.

Part 2

Time, Frequency and Phase Estimation

8

Non-Parametric Estimation of the Periodic Signal Parameters in the Frequency Domain

Dušan Agrež
University of Ljubljana,
Faculty of Electrical Engineering
Slovenia

1. Introduction

Parameters estimations of periodic signals $g(t)$, where frequency of the investigated component is the key parameter, play a fundamental role in a variety of applications: impedance measurement, power quality estimation, radar, A/D testing, etc. The problem of evaluating the spectral performance of a given periodic signal reduces to the parameter estimation of each spectral component (frequency, amplitude, and phase) in the presence of noise. Estimation methods can be classified as parametric (D'Antona & Ferrero, 2006) and nonparametric (Agrež, 2002). Parametric methods are model-based and have very good selectivity and statistical efficiency, but require computationally intensive algorithms and very good 'model agreement' with a real multi-component signal. For this reason, such methods are unsuitable for many estimation problems. A better approach is to use non-parametric methods, which estimate the spectral parameters of interest by evaluating first the discrete Fourier transform (DFT) of the signal and then the parameters of the particular component. As we are dealing with periodic signals, the integral frequency transformation with the kernel $e^{-j2\pi ft}$ is, in principle, the best approximation to periodicity of the signal. Analysis of the frequency spectrum provides the opportunity to see systematic periodicities in the presence of the reduced random noise by integration. Many of these estimations are based on coherent sampling; that is, on the accurate synchronization of the signal and the sampling rate, and on the collection of a number of samples belonging to an integer number of the signal periods. However, the normal situation for signal parameter estimation is non-coherent, or quasi-coherent sampling, and in such a sampled signal there can also be spurious components. When failing to observe an integer number of periods of even a single tone, the tone energy is spread over the whole frequency axis, and the leakage from neighboring components can significantly bias estimations of the component parameters.

There are two fundamental principles that restrict estimations: the time-frequency uncertainty principle $\Delta T\,\Delta W \geq 1/\pi$ (Gabor, 1946), and the principle of the limited changes of signals. The first, where ΔT and ΔW are the effective widths of lobes in the time and frequency domains (1), is a generalization of the Heisenberg uncertainty principle.

$$(\Delta T)^2 = 4\left[\underbrace{\frac{\int_{-\infty}^{\infty} t^2 g^2(t)\,dt}{\int_{-\infty}^{\infty} g^2(t)\,dt}}_{\text{The normalized moment of inertia about the center of gravity of the signal distribution.}} - \underbrace{\left(\frac{\int_{-\infty}^{\infty} t g^2(t)\,dt}{\int_{-\infty}^{\infty} g^2(t)\,dt}\right)^2}_{\text{The normalized first moment – center of gravity of the signal distribution.}}\right] \tag{1a}$$

$$(\Delta W)^2 = 4\frac{\int_{-\infty}^{\infty} f^2 \left|G(f)\right|^2 df}{\int_{-\infty}^{\infty} \left|G(f)\right|^2 df} \tag{1b}$$

In words, ΔT and ΔW cannot simultaneously be arbitrary small. In relation to measurements this is interpreted to imply that the uncertainty in the determination of a frequency, is of the order of magnitude of the reciprocal of the time taken to measure it.

The second principle, that of the limited changes of signals, limits the design of the signal shape. The more smoothly and slowly a function changes, the more rapidly its transformation changes and vice versa (Seibert, 1986). In practice this means that the spectrum of the signal should essentially vanish for frequencies greater than some frequency f_{max}, and that the tails of the pulse in the time domain must die sufficiently rapidly that the tail of a large pulse will not seriously distort another smaller pulse at an adjacent time instant. The quantitative expression of this principle can be derived from Parseval's theorem and the differentiation properties of the Fourier transformation $\mathrm{d}g(t)/\mathrm{d}t \Leftrightarrow \mathrm{j}2\pi f G(f)$:

$$\int_{-\infty}^{\infty} \left|\frac{\mathrm{d}^{\mathrm{n}} g(t)}{\mathrm{d}t^{\mathrm{n}}}\right|^2 \mathrm{d}t = (2\pi)^2 \int_{-\infty}^{\infty} f^{2n} \left|G(f)\right|^2 \mathrm{d}f \tag{2}$$

Thus, if all the derivatives of the signal $g(t)$ through the $(n-1)^{\mathrm{st}}$ are square-integrable, but the n^{th} is not, we may in general conclude that its spectrum $G(f)$ vanishes faster than $|f|^{-n+1/2}$, but not faster than $|f|^{-n-1/2}$.

Both fundamental principles limit the accuracy of parameter estimations and depend upon the measurement time. Here this is taken to mean the relative time to measure a periodic signal, or the number of repetitions of the periodicity in it:

$$\theta = \frac{T_{\mathrm{meas}}}{T_{\mathrm{period}}} = T_{\mathrm{meas}} \cdot f = \frac{f}{\Delta f} \tag{3}$$

where the measurement time determines the basic frequency resolution in the frequency domain $\Delta f = 1/T_{\mathrm{meas}}$.

A finite time of measurement is a source of dynamic errors, which are shown as leakage parts of the measurement window spectrum, convolved on the spectrum of the measured-sampled signal (Fig. 1). The sampled analogue multi-frequency signal $g(t)$, by $f(\mathrm{sampling}) = 1/\Delta t$, can be written as follows:

$$g(k\Delta t)_N = w(k)\cdot\sum_{m=0}^{M} A_m \sin(2\pi f_m k\Delta t + \varphi_m), \qquad k = 0, 1, \ldots, N-1 \tag{4}$$

where f_m, A_m, and φ_m are frequency, amplitude, and phase of one component with index m among $M+1$, respectively. Index k is the current time index of the successive samples Δt apart. Tones of the sampled signal do not generally coincide with the basic set of the periodic components of the DFT, which is the most well-known, non-parametric method for frequency decomposition of signals (Harris, 1978). Using N samples of the signal (4), the DFT at the spectral line i is given by:

$$G(i) = -\frac{j}{2}\sum_{m=0}^{M} A_m\left[W(i-\theta_m)e^{j\varphi_m} - W(i+\theta_m)e^{-j\varphi_m}\right], \tag{5}$$

where $W(*)$ is the spectrum of the used window $w(k)$, and θ_m is the signal component frequency divided by the frequency resolution $\Delta f = 1/(N\Delta t)$, and can be written in two parts:

$$\theta_m = \frac{f_m}{\Delta f} = i_m + \delta_m \qquad -0.5 < \delta_m \le 0.5 \tag{6}$$

where i_m is an integer value and the displacement term δ_m is caused by the non-coherent sampling. The DFT coefficients surrounding one signal component are due to both the short-range leakage and the long-range leakage contributions from the second term of the investigated component, and from both terms of other components (7) (Fig. 1).

$$\left|G(i_m)\right| = \frac{A_m}{2}\left|W(\delta_m)e^{j\varphi_m} - W(2i_m+\delta_m)e^{-j\varphi_m}\right| + \sum_{k=0,k\neq m}^{M}\left|\Delta(i_k)\right| \tag{7}$$

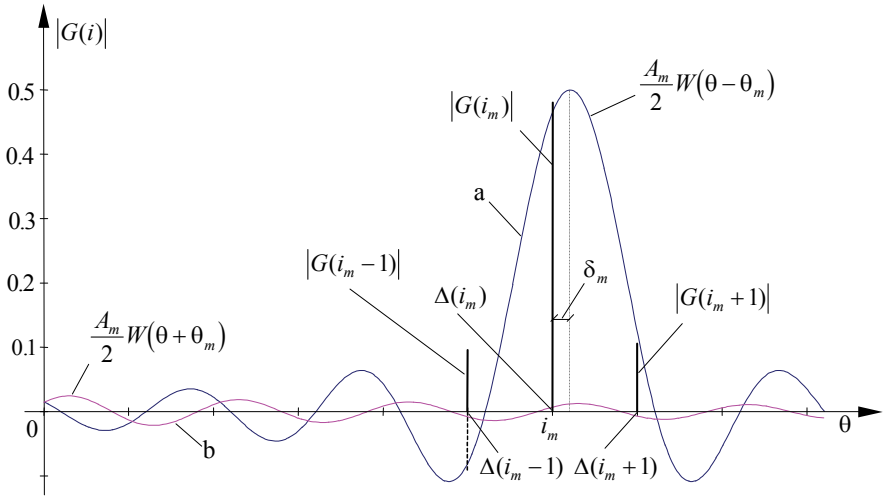


Fig. 1. The short-range leakage influences (a) and long-range leakage influences (b) on the amplitude DFT coefficients (rectangular window; $\theta_m = 6.3$)

The long-range leakage contributions can be reduced in several ways: by increasing the measurement time, by using windows with a faster reduction of the side lobes than with a rectangular window (like the Hann window, Rife-Vincent windows, Dolph-Chebychev windows, etc., Fig. 2), or by using the multi-point interpolated DFT algorithms and a window with known behavior of the spectrum (Agrež, 2002). For the sake of analytical simplicity, cosine-class windows are frequently used (Belega & Dallet, 2009; Novotný & Sedláček, 2010). The three basic classes of cosine windows were defined RV1, RV2, and RV3. For analyses, the first two classes are interesting.

Windows of the class RV1 (Fig. 2: curves a, b, and c) are designed for maximization of the window spectrum side-lobes fall-off θ^{-b}, based on the number of the time domain window derivatives zeroes at the window ends (Novotný & Sedláček, 2010):

$$w(k) = \sum_{l=0}^{b-1} (-1)^l a_{l,1} \cdot \cos\left(l \frac{2\pi}{N} \cdot k \right), \; k = 0, \ldots, N-1 \tag{8}$$

When the order b is 1 (RV1-1), the coefficient $a_{0,1}$ is 1 and the equation (8) gives the rectangular shape. If b is 2 ($\mathrm{RV1-2}: a_{0,1} = 1/2$, $a_{1,1} = 1/2$) we get the Hann window. Higher values of b ($\mathrm{RV1-4}: a_{0,1} = 10/32$, $a_{1,1} = 15/32$, $a_{2,1} = 6/32$, $a_{3,1} = 1/32$) expand the window transform main-lobe and reduce the spectral leakage.

Windows of the class RV2 (Fig. 2: curves d and e) are designed for minimization of the window spectrum main-lobe width, for a given maximum level of the side-lobes relative magnitude $\left(A_{\text{side-lobe}}/A(0)\right)_{\max} = 10^{-d}$, where d is the exponent of the damping. They are the Taylor approximation to the Dolph-Chebychev windows and give good results when spectral components are very close (Andria et al., 1989).

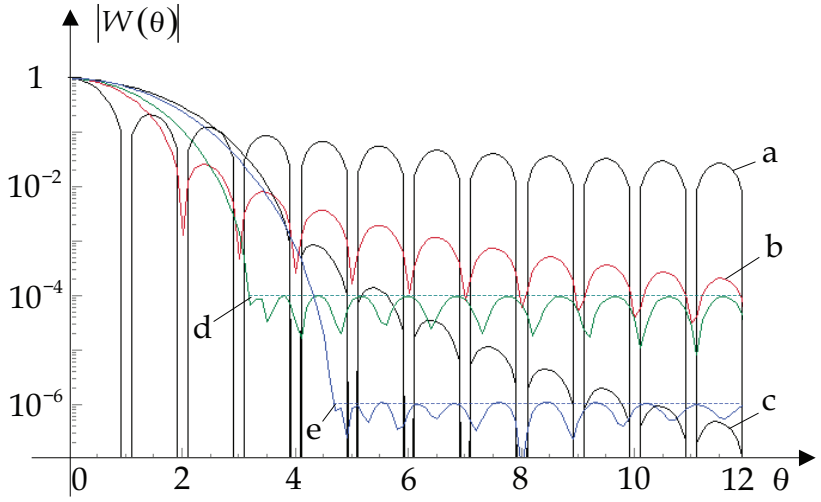


Fig. 2. Spectra shapes of the windows: a – the rectangular window (RV1-1), b – the Hann window (RV1-2), c – the RV1-4 window, d – the RV2 window with $d = 4$ (RV2-4), e – the RV2 window with $d = 6$ (RV2-6)

2. Parameters estimations

It is important how the used window works in the estimation. Especially in the first step if we have an iterative procedure of estimation. Estimation of the first step has the non-parametric nature since there is no information about the signal at the beginning.

2.1 Frequency estimation

Parameters of the measurement component can be non-parametric estimates by means of the interpolation. From the comparative study (Schoukens et al., 1992) it can be concluded that the key for estimating the three basic parameters is in determining the position of the measurement component $\delta_m = \theta_m - i_m$, between DFT coefficients $G(i_m)$ and $G(i_m+1)$, surrounding the component m (Fig. 1.). Estimation can be done by multi-point estimations (Agrež, 2002) and using windows with known spectra, like the Hann window:

$$\left|W_{\mathrm{H}}(\theta)\right|_{N>>1} = \frac{\left|\sin(\pi\theta)\right|}{\left|2\pi\theta\left(1-\theta^2\right)\right|} \qquad \text{in} \quad \left|G(i_m)\right| \doteq \frac{A_m}{2}\left|W(\delta_m)\right| \tag{9a}$$

$$\text{for 2-points estimation:} \quad {}_2\delta_m \doteq s\frac{2\left|G(i_m+s)\right| - \left|G(i_m)\right|}{\left|G(i_m)\right| + \left|G(i_m+s)\right|} \tag{9b}$$

where $s = \mathrm{sign}(\delta_m)$ is the sign of displacement and can be estimated by the difference of the phase DFT coefficients: $s = sign\left(\left|arg\left[G(i_m)\right] - arg\left[G(i_m+1)\right]\right| - \pi/2\right)$.

The three largest, local DFT coefficients can be used in the three-point interpolation and in this case, long-range leakage contributions can be considered. Portions $\left|\Delta(i)\right|$ (7) of the long-range leakage tails have the following properties: they decrease with increasing frequency and they change sign at successive coefficients $\left|G(i)\right|$, if they have a sine function in the kernel $\left(\sin(\pi(i+\delta_m)) = -\sin(\pi(i\pm1+\delta_m))\right)$. For example, the rectangular window, the Hann window, and Rife-Vincent Class I windows satisfy this condition. For θ_m that is large enough, the long-range leakage influence can be approximated to $\left|\Delta(i_m-1)\right| \approx \left|\Delta(i_m)\right| \approx \left|\Delta(i_m+1)\right|$, so that the ratio of coefficients can be expressed as:

$$\begin{aligned} {}_3\alpha_m &= \frac{\left|G(i_m)\right| + \left|G(i_m-1)\right|}{\left|G(i_m)\right| + \left|G(i_m+1)\right|} = \\ &= \frac{\left|W(\delta_m)\right| + \left|\Delta(i_m)\right| + \left|W(1+\delta_m)\right| - \left|\Delta(i_m-1)\right|}{\left|W(\delta_m)\right| + \left|\Delta(i_m)\right| + \left|W(1-\delta_m)\right| - \left|\Delta(i_m+1)\right|} \approx \frac{\left|W(\delta_m)\right| + \left|W(1+\delta_m)\right|}{\left|W(\delta_m)\right| + \left|W(1-\delta_m)\right|} \end{aligned} \tag{10}$$

and the displacement term as:

$${}_3\delta_m \cong 2\frac{1-{}_3\alpha_m}{1+{}_3\alpha_m} = 2\frac{\left|G(i_m+1)\right| - \left|G(i_m-1)\right|}{\left|G(i_m-1)\right| + 2\left|G(i_m)\right| + \left|G(i_m+1)\right|} \tag{11}$$

The numerators and the denominators in (9) and (11) have the form where the amplitude DFT coefficients are added with suitable weights. The form of the denominator in (11) $\left|G_{\mathrm{H}}(i-1)\right| + 2\left|G_{\mathrm{H}}(i)\right| + \left|G_{\mathrm{H}}(i+1)\right|$ is very characteristic, and looks like the form in the

construction of the Hann window spectrum with the Dirac delta function $D(*)$, and the spectrum of the rectangular window (Harris, 1978):

$$W_{\text{Hann.}}(\theta)=\left(\frac{1}{2}D(i)-\frac{1}{4}D(i-1)-\frac{1}{4}D(i+1)\right)\otimes W_{\text{rect.}}(\theta) \tag{12}$$

but instead of the rectangular window, the Hann window can be used:

$$W(\theta)=\left(D(i-1)+2D(i)+D(i+1)\right)\otimes W_{\text{Hann.}}(\theta) \tag{13}$$

From the point of view of leakage, the denominator is a sum of the weighted leakages. We can get the weights with a triple subtraction of the long-range leakage tails:

$$\begin{gathered}\left|\Delta(i_m-1,i_m)\right|=\left|\Delta(i_m-1)\right|-\left|\Delta(i_m)\right|;\quad \left|\Delta(i_m,i_m+1)\right|=\left|\Delta(i_m)\right|-\left|\Delta(i_m+1)\right| \\ \begin{aligned}\left|\Delta(i_m-1,i_m,i_m+1)\right| &= \left|\Delta(i_m-1,i_m)\right|-\left|\Delta(i_m,i_m+1)\right| \\ &= \left|\Delta(i_m-1)\right|-2\left|\Delta(i_m)\right|+\left|\Delta(i_m+1)\right| << \left|\Delta(i_m)\right|\end{aligned}\end{gathered} \tag{14}$$

The numerator is a subtraction of the sum of the first two, from the sum of the last two DFT coefficients $\left(\left|G(i_m)\right|+\left|G(i_m+1)\right|\right)-\left(\left|G(i_m-1)\right|+\left|G(i_m)\right|\right)=\left|G(i_m+1)\right|-\left|G(i_m-1)\right|$. In this case the long-range leakage tails are also reduced.

$$\left(\left|\Delta(i_m)\right|-\left|\Delta(i_m+1)\right|\right)-\left(-\left|\Delta(i_m-1)\right|+\left|\Delta(i_m)\right|\right)=\left|\Delta(i_m-1)\right|-\left|\Delta(i_m+1)\right| \tag{15}$$

It is appropriate to form multi-point interpolations on an odd number of coefficients, in order to have symmetry around the largest local coefficient $\left|G(i_m)\right|$. In a five-point interpolation with the Hann window, similar averages are used as in the three-point interpolation. The quotient is used to eliminate the amplitude influence of the investigated component.

$${}_5\delta_m=3\cdot\frac{2\left[\left|G(i_m+1)\right|-\left|G(i_m-1)\right|\right]+s\cdot\left[\left|G(i_m+2)\right|+\left|G(i_m-2)\right|\right]}{6\left|G(i_m)\right|+4\left[\left|G(i_m+1)\right|+\left|G(i_m-1)\right|\right]+s\cdot\left[\left|G(i_m+2)\right|-\left|G(i_m-2)\right|\right]} \tag{16}$$

An estimation of the periodic parameter by the interpolation of the DFT gives the same effect as the reduction of spectrum tails. The meaning of the interpolation is the weighted summation of the amplitude coefficients, or better, symmetrical subtraction of the successive adjacent leakage parts of the window spectrum (14). The idea for long-range leakage reduction by summation of the adjacent weighted DFT coefficients, is at the core of the construction of the cosine class windows. Weights for forming the Hann window and the Rife-Vincent Class I windows from the rectangular window, are obtained by repeated convolution of the two-point weight pairs (1, 1), that is by repeated subtraction of the neighboring pairs of the spectrum leakage tails. Binomial weights $\binom{2r}{j}$ ($r=1, 2, \ldots$ is a number of coefficients of one half; $j=0,\ 1,\ \ldots,\ 2r$) can be obtained from a Pascal triangle.

The displacement estimations with the multi-point ($\eta=2r+1=3,\ 5,\ 7,\ \ldots$) interpolations of the DFT using the Hann window can be written as:

$$ {}_{2r+1}\delta_m \cong (r+1)\cdot \frac{K_{n_1}\left[\left|G(i_m+1)\right|-\left|G(i_m-1)\right|\right]+\ldots+(-1)^l sK_{n_l}\left[\left|G(i_m+l)\right|+\left|G(i_m-l)\right|\right]}{K_{d_1}\left|G(i_m)\right|+K_{d_2}\left[\left|G(i_m+1)\right|+\left|G(i_m-1)\right|\right]+\ldots+(-1)^l sK_{d_l}\left[\left|G(i_m+l)\right|-\left|G(i_m-l)\right|\right]} $$

$$ {}_{\eta}\delta_m \cong \left(\frac{\eta+1}{2}\right)\cdot\frac{d_\eta}{n_\eta} \tag{17} $$

where l ($l=1,2,\ldots\le r$) is the current index. The first term of the numerator in (17) is a difference of the side coefficients around the largest one, and the remaining terms are sums of the symmetrical pairs of coefficients ($\left|G(i_m\pm l)\right|$). All terms $[*]$ are weighted by $K_{n_l}=\binom{2(r-1)}{(r-l)}-\binom{2(r-1)}{(r-l-2)}$. The signs of weights alter successively. The largest coefficient in the denominator is weighted by $K_{d_1}=\binom{2r}{r}$, the sum of the first side coefficients by $K_{d_2}=\binom{2r}{(r-1)}$ and, the differences of the symmetrically located coefficients are weighted by $K_{d_{l+1}}=\binom{2r}{(r-l)}$. The signs of weights also alter successively $(-1)^l$.

2.1.1 Reduction of systematic error

The results of the simulations, where the relative frequency was changed, show that the systematic contribution of the error – the estimation bias - decreases (Fig. 3.), while the influence of noise on estimations (σ_{θ_m}) increases with the number of interpolation points (Fig. 4.).

The errors of the frequency estimations $E(\theta)=(i+\delta)-\theta^*$ (θ^* is the true value of the relative frequency) for one sine component in the signal have been checked with a double scan varying both frequency and phase ($A_m=1$, $N=1024$, $1\le\theta\le 6$, $\Delta\theta=0.001$ and $-\pi/2\le\varphi\le\pi/2$, $\Delta\varphi=\pi/180$). The absolute maximum values of the errors (from 181 iterations) at the given relative frequency were compared for the multi-point DFT interpolations using the Hann window (Fig. 3).

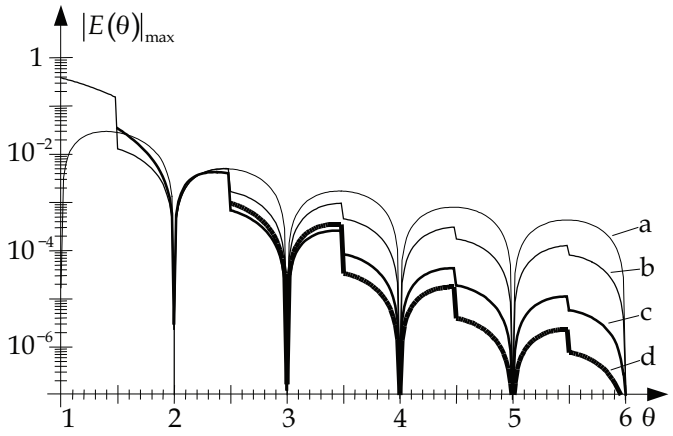


Fig. 3. Maximal errors of frequency estimation with the multi-point interpolations of the DFT for the Hann window (a: two-point interpolation; b: three-point interpolation; c: five-point interpolation; d: seven-point interpolation)

2.1.2 Uncertainty of the frequency estimations

Distributions of errors ($E = |G(i)|_{\text{noise}} - |G(i)|_{\text{noiseless}}$) of the largest amplitude DFT coefficients have very similar (Gaussian) shapes with almost equal standard deviations $\sigma_{|\text{DFT}|}(i_m) \approx \sigma_{|\text{DFT}|}(i_m \pm 1,2,.) = \sigma_{|\text{DFT}|}$, if the time domain noise with standard deviation σ_t and a mean value $u(t) \to 0$, is statistically independent of the signal and sampling process (18). In other words, if in the first approximation only the quantization noise is considered, the sampling frequency should be suitably larger than the frequency of the highest frequency component (f_M) $f_S/f_M >> 2$ and the effective number of bits ($ENOB = b$) of the A/D conversion suitably high enough $b \geq 3$, or $SNR \geq 20\text{dB}$ (Widrow, Kollar. 2008). The signal to noise ratio in the time domain is defined as $SNR = A_m^2 \Big/ \left(2\sigma_t^2\right)$ ($SNR/\text{dB} = 10\log SNR$), or expressed with an effective number of bits $SNR = 3 \Big/ 2 \cdot \left(2^b\right)^2$, where the rectangular distribution of the quantization errors is taken into account $\sigma_t = A_m \Big/ \left(2^b\right) \cdot 1/\sqrt{3}$.

$$\sigma_{|\text{DFT}|} = \sigma_t \ \frac{1}{N\sqrt{2}} \sqrt{\sum_{k=0}^{N-1} w^2(k)} \tag{18}$$

The "absolute" form of the standard deviation (18) is usually related to the values of the DFT coefficients of interest. In coherent sampling the largest local amplitude DFT coefficient is equal to $|G(i_m)| = A_m/2 \cdot \sum w(k)/N$, where $\sum w(k)/N \leq 1$ represents the normalized peak signal gain of the window $w(k)$ (Solomon 1992). The relative form of the standard deviation can be written as:

$$\sigma^*_{|\text{DFT}|} = \frac{\sigma_{|\text{DFT}|}}{|G(i_m)|} = \frac{\sigma_x}{A_m}\sqrt{2}\frac{\sqrt{\sum_{k=0}^{N-1} w^2(k)}}{\sum_{k=0}^{N-1} w(k)} = \frac{\sigma_x}{A_m}\frac{\sqrt{2}}{\sqrt{N}}\sqrt{ENBW} \tag{19}$$

The root of the equivalent noise bandwidth $ENBW$ (Harris, 1978) is a factor determining the size of the standard deviation when using different windows: $ENBW > ENBW_{\text{rect.}} = 1$.

Distortions of the DFT coefficients and their number in an interpolation have a significant influence on the uncertainty of the displacement estimation. The standard deviation of δ_m, as a dependent quantity, can generally be expressed as (Joint Committee for Guides in Metrology [JCGM], 2008):

$$\sigma^2_{\delta_m} = \sum_{p=i_m-r}^{p=i_m+r} \left(c_\delta(p) \cdot \sigma_{|G(p)|}\right)^2 + 2 \sum_{p=i_m-r}^{p=i_m+r-1} \sum_{v=p+1}^{v=i_m+r} \left(r_C\left(|G(p)|, |G(v)|\right) \cdot c_\delta(p) \cdot c_\delta(v) \cdot \sigma_{|G(p)|} \cdot \sigma_{|G(v)|}\right) \tag{20}$$

where $c_\delta(p) = \partial \delta_m \big/ \partial |G(p)|$ ($p = i_m - r, \ldots, i_m + r$) is the sensitivity coefficient associated with the amplitude coefficient $|G(p)|$, and $r_C\left(|G(p)|, |G(v)|\right)$ ($p \neq v$) is the correlation coefficient. In the case of the Hann window, two successive amplitude coefficients have the correlation factor $r_C\left(|G(p)|, |G(p+1)|\right) = 2/3$, and amplitude coefficients with a current index of two apart, have $r_C\left(|G(p)|, |G(p+2)|\right) = 1/6$. Other correlation coefficients are zero.

As the standard deviations of the amplitude coefficients are almost equal $\sigma_{|G(p)|} \cong \sigma_{|G(v)|} = \sigma_{|\mathrm{DFT}|}$, it is possible to formulate the expression for the standard deviation of displacement. For the three-point interpolation using the Hann window (11), it can be expressed as:

$$
\begin{aligned}
{}_3\sigma^2_{\delta_m} = \sigma^2_{|\mathrm{DFT}|} \Big(& \left(c^2_\delta(i_{m-1}) + c^2_\delta(i_m) + c^2_\delta(i_{m+1})\right) \\
& + \frac{4}{3}\left(c_\delta(i_m - 1)\cdot c_\delta(i_m) + c_\delta(i_m)\cdot c_\delta(i_m + 1)\right) \\
& + \frac{2}{6}\left(c_\delta(i_m - 1)\cdot c_\delta(i_m + 1)\right)\Big)
\end{aligned}
\tag{21}
$$

The same mathematical procedure can be used for other higher multi-point interpolations (17).

$$
\left({}_\eta\sigma_{\delta_m} \Big/ \sigma_{|\mathrm{DFT}|}\right)^2 = \sum_{p=i_m-r}^{p=i_m+r} c^2_\delta(p) + \frac{4}{3}\cdot \sum_{p=i_m-r}^{p=i_m+r-1} c_\delta(p)\cdot c_\delta(p+1) + \frac{1}{3}\cdot \sum_{p=i_m-r}^{p=i_m+r-2} c_\delta(p)\cdot c_\delta(p+2) \tag{22}
$$

Sensitivity coefficients $c(p)$ have forms such as $c(p) = \left(\left(d_\eta\right)' \cdot n_\eta - \left(n_\eta\right)' \cdot d_\eta\right) \Big/ n^2_\eta$, where d_η, n_η are the denominator and the numerator of fraction (17), respectively. Both are sums of the weighted amplitude DFT coefficients, which change with relative displacement - the short-range leakage influences. For this reason, the standard deviations of displacements change their values periodically $-0.5 < \delta_m \le 0.5$ (Fig. 4).

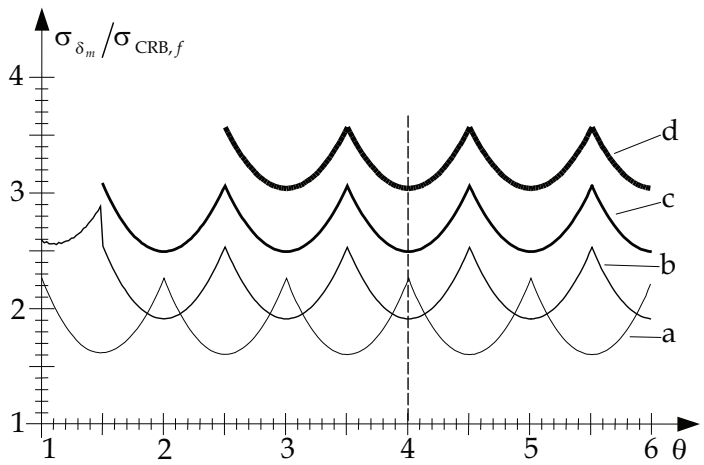


Fig. 4. Standard deviations of the displacement estimation related to the CRB standard deviation for the frequency estimation (a: two-point interpolation; b: three-point interpolation; c: five-point interpolation; d: seven-point interpolation)

The standard deviation of the displacement estimation is related to the absolute form of the standard deviation of the amplitude DFT coefficient with a suitable factor

$\sigma_{\delta_m} = \sigma_{|\mathrm{DFT}|} \cdot {}_{\eta} R_{\delta}$ according to (22). If one wants to compare it with the unbiased Cramér-Rao lower bound (CRB) for the estimation of the frequency (Petri, 2002) the relationship has to be reexpressed:

$$\sigma_{\delta_m} = \frac{\sigma_{|\mathrm{DFT}|}}{\left|G(i_m)\right|} \cdot \left(\left|G(i_m)\right| \cdot {}_{\eta} R_{\delta}\right) = \frac{1}{\sqrt{SNR}} \frac{\sqrt{ENBW}}{\sqrt{N}} \cdot \left(\left|G(i_m)\right| \cdot {}_{\eta} R_{\delta}\right) \geq \frac{1}{\sqrt{SNR}} \frac{1}{\sqrt{N}} \frac{\sqrt{3}}{\pi} = \sigma_{\mathrm{CRB},f} \quad (23)$$

This form is larger than $\sigma_{\mathrm{CRB},f}$ for the frequency estimation taking into account all measurement information (Fig. 4).

Errors in relative frequency estimations with different numbers of interpolation points have normal distributions. The standard deviation of the three-point frequency estimation, which has the lowest standard deviation in the vicinity of the integer values of the relative frequency, is about 2.2 times higher than $\sigma_{\mathrm{CRB},f}$ (Fig. 4: curve b). The lowest value ($_3\sigma_{\delta_m} / \sigma_{\mathrm{CRB},f} \approx 1.9$) is attained at $\delta_m \approx 0$, and the highest ratio ($_3\sigma_{\delta_m} / \sigma_{\mathrm{CRB},f} \approx 2.55$) is at the worst cases of the non-coherent sampling ($\delta_m \approx 0.5$). A two-point interpolation is worse around $\delta_m \approx 0$ ($_2\sigma_{\delta_m} / \sigma_{\mathrm{CRB},f} \approx 2.26$), but it is superior in the interval $0.15 \leq |\delta_m| \leq 0.5$ ($_2\sigma_{\delta_m} / \sigma_{\mathrm{CRB},f} \geq 1.59$).

2.1.3 A trade-off between bias and uncertainty

If we reduce the leakage tails, or systematic errors by the interpolation, we apparently widen the estimation main-lobe ΔW_{est} ($ENBW > 1$), and the noise in the estimation increases in comparison to the CRB. For example, the noise of the cosine windows increases $ENBW\left(\cos^2(X)\right) / ENBW\left(\cos^4(X)\right) \approx 1.5/1.94 \approx 1/1.29$, while the side-lobes levels decrease $SL\,fall\left(\cos^2(X)\right) / SL\,fall\left(\cos^4(X)\right) \approx -18\mathrm{dB} / -30\mathrm{dB}$ (Harris, 1978). At the same time, the systematic errors ($|E_{\theta}|_{\max}$) decrease with increasing numbers of points. Increasing the number of the used DFT coefficients is reasonable until the systematic error drops under the noise error. After this point, by increasing the relative frequency θ_m, or with spacing between the two frequency components ($\propto 2 \cdot \theta_m$), it is logical to decrease the number of interpolation points.

The criterion for selecting one of the algorithms could be the minimum common uncertainty of the estimation considering both contributions:

$$\sigma_{\Sigma} = \sqrt{\left(\frac{\left|E(\theta_m)\right|_{\max}}{\sqrt{2}}\right)^2 + \sigma_{\theta_m}^2} \quad \rightarrow \quad \min \quad (24)$$

The effective value of the systematic contribution is obtained by dividing the maximal error by the square root of two, since systematic errors are phase dependent with a sine like shape.

The borders of relative frequency where one interpolation can pass over another depend upon the number b of bits of the A/D converter. With a 10-bit A/D converter ($SNR \approx 62\mathrm{dB}$), which is frequently used in industrial environments, it is convenient to use the three-point DFT interpolation with the Hann window, in the interval $1 < \theta < 2.2$ and

from $6.7 < \theta$ onward (Fig. 5). Between values 2.2 and 6.7 of the relative frequency, it is better to use the five-point interpolation (or even the seven-point interpolation in intervals $[3.5, 3.8]$ and $[4.3, 4.5]$).

Multi-point interpolations present worse results: at lower θ owing to systematic error ("window width"), and at higher values of θ owing to the larger noise sensibility. The solid line at the top of Fig. 5 shows where different multi-point interpolations can be used to achieve the best results of the one-component frequency estimation.

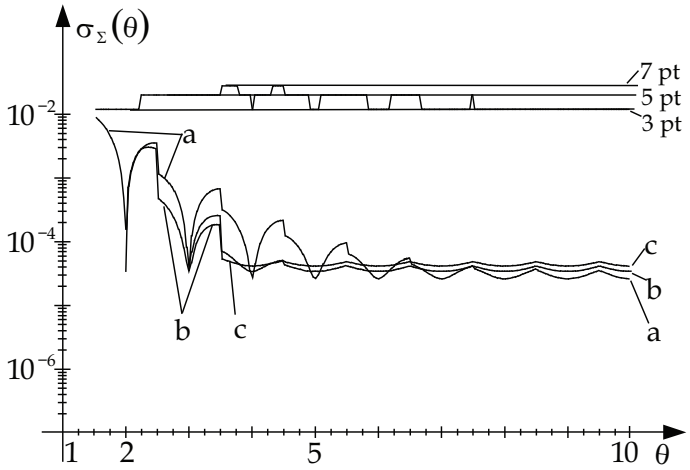


Fig. 5. The use of multi-point DFT interpolations for a 10-bit A/D converter (a: three-point interpolationb: five-point interpolationc: seven-point interpolation)

2.2 Amplitude estimation

From the behavior of the systematic error of the frequency estimation (Fig. 3), it can be concluded that it is better to use the Hann, or some higher order cosine window for the estimation, if the window spectrum is analytically known. When the displacement δ_m for the specific component is determined, it is easy to get the amplitude using the Hann window (9a) and neglecting the long-range contribution $|\Delta(i_m)|$ in (7):

$$A_m = 2\left|\frac{2\pi\delta_m\left(1-\delta_m^2\right)}{\sin(\pi\delta_m)}\right||G(i_m)| \tag{25}$$

As in the case of the frequency estimation, with the summation of the DFT coefficients, we subtract the long-range leakage tails and reduce their influences. We get the weights for the three-point summation with the triple subtraction of the long-range leakage tails (14).

$$|\Delta(i_m-1)|-2|\Delta(i_m)|+|\Delta(i_m+1)| << |\Delta(i_m)| \tag{26}$$

In this manner, we can get the amplitude of the signal by summing the largest three local DFT coefficients around the signal component following the result of (26):

$$\left[\left|G(i_m-1)\right|+2\left|G(i_m)\right|+\left|G(i_m+1)\right|\right]=\frac{A_m}{2}\left[\left|W(1+\delta_m)\right|\pm\left|\Delta(i_m-1)\right|+\right.$$
$$\left.+2\left|W(\delta_m)\right|\mp 2\left|\Delta(i_m)\right|+\left|W(1-\delta_m)\right|\pm\left|\Delta(i_m+1)\right|\right]$$

$$A_m\cong 2\cdot\frac{\left[\left|G(i_m-1)\right|+2\left|G(i_m)\right|+\left|G(i_m+1)\right|\right]}{\left|W(1+\delta_m)\right|+2\left|W(\delta_m)\right|+\left|W(1-\delta_m)\right|} \tag{27}$$

Using the Hann window:

$$\left|W_{\mathrm{H}}(\delta_m)\right|\cong\frac{\sin(\pi\delta_m)}{2\pi\delta_m\left(1-\delta_m^2\right)}\text{ and }\left|W_{\mathrm{H}}(1+s\delta_m)\right|\cong\frac{\sin(\pi\delta_m)}{2\pi\delta_m(1+s\delta_m)(2+s\delta_m)} \tag{28}$$

$$\left|W(1+\delta_m)\right|+2\left|W(\delta_m)\right|+\left|W(1-\delta_m)\right|=\frac{\sin(\pi\delta_m)}{2\pi\delta_m}\frac{12}{\left(1-\delta_m^2\right)\left(4-\delta_m^2\right)}, \tag{29}$$

the amplitude estimation with the three-point interpolation ($_3A_{m\mathrm{H}}$) can be expressed as follows:

$${}_3A_{m\mathrm{H}}\cong\frac{\pi\delta_m}{\sin(\pi\delta_m)}\frac{\left(1-\delta_m^2\right)\left(4-\delta_m^2\right)}{3}\cdot\left[\left|G(i_m-1)\right|+2\left|G(i_m)\right|+\left|G(i_m+1)\right|\right] \tag{30}$$

We can use the same procedure for the five-point interpolation with ten subtractions of the tails. In the first step of the procedure we do four subtractions of the adjacent tails $\left|\Delta(i_m-2,i_m-1)\right|$, …, $\left|\Delta(i_m+1,i_m+2)\right|$ as in (14), or summations of the DFT coefficients, then three subtractions of the obtained and reduced tails as in (26), and so on. After rearrangement the amplitude can be expressed with the weighted five largest coefficients:

$$\begin{aligned}{}_5A_{m\mathrm{H}}&=\frac{\pi\delta_m}{\sin(\pi\delta_m)}\frac{\left(1-\delta_m^2\right)\left(4-\delta_m^2\right)\left(9-\delta_m^2\right)}{90}\cdot\\&\cdot\left[6\left|G(i_m)\right|+4\left(\left|G(i_m+1)\right|+\left|G(i_m-1)\right|\right)+\left|\left|G(i_m+2)\right|-\left|G(i_m-2)\right|\right|\right]\end{aligned} \tag{31}$$

In the last term of the equation (31), the absolute value of the difference is used, since one of the coefficients $\left|G(i_m-2)\right|$ or $\left|G(i_m+2)\right|$ drops out of the spectrum main lobe ($4\Delta f$ wide in the Hann case) and gets a negative sign.

The same procedure can be used for the seven-point interpolation:

$$\begin{aligned}{}_7A_{m\mathrm{H}}&=\frac{\pi\delta_m}{\sin(\pi\delta_m)}\frac{\left(1-\delta_m^2\right)\left(4-\delta_m^2\right)\left(9-\delta_m^2\right)\left(16-\delta_m^2\right)}{5040}\cdot\\&\cdot\left[20\cdot\left|G(i_m)\right|+15\cdot\left(\left|G(i_m+1)\right|+\left|G(i_m-1)\right|\right)+\right.\\&\left.+6\cdot\left|\left|G(i_m+2)\right|-\left|G(i_m-2)\right|\right|-\left|\left|G(i_m+3)\right|-\left|G(i_m-3)\right|\right|\right]\end{aligned} \tag{32}$$

The relative error $e(A)=\left(A/A^*-1\right)$ ($A^*=1$ is the true value of the amplitude) drops with increasing relative frequency and with the number of the interpolation points (Fig. 6: ${}_0e$ - the amplitude is estimated only with the largest coefficient, ${}_1e$ - estimation with (25), ${}_3e$ - estimation with (30), etc; The same testing conditions as for Fig. 3). Comparing figures 6 and 7 shows the importance of the frequency estimation accuracy. If we know the value of the

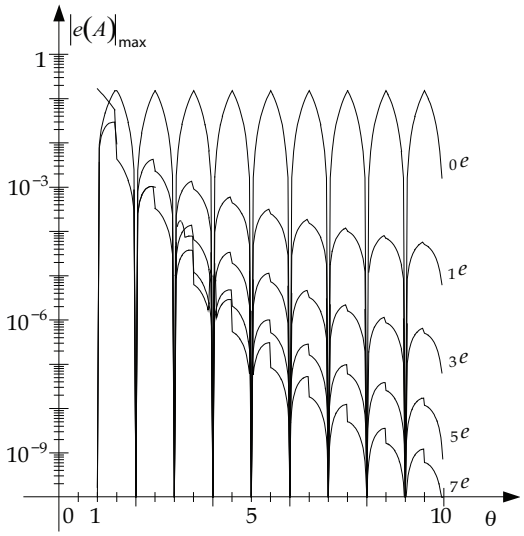


Fig. 6. Maximal relative values of errors of the amplitude estimation with the multi-point DFT interpolations with the Hann window (θ is known)

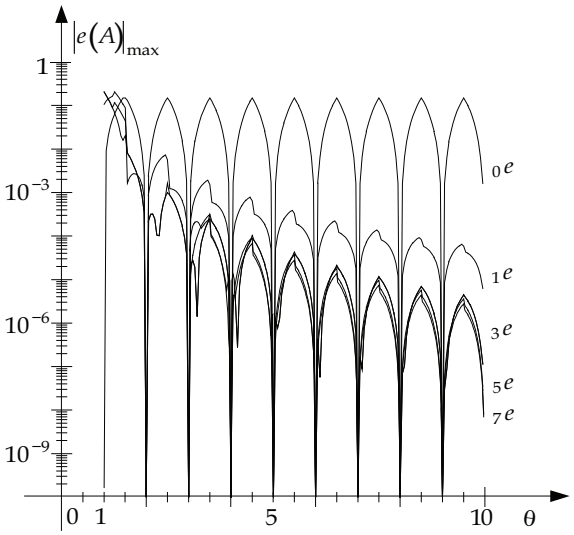


Fig. 7. Maximal relative values of errors of the amplitude estimation with the multi-point DFT interpolations with the Hann window (θ is obtained with the three-point int. (11))

frequency on the three-point interpolation accuracy level, then the amplitude estimation is reasonable with the three-point interpolation. The accuracy of the amplitude estimation can be improved, if the frequency is better estimated (e.g., by the multipoint interpolations).

2.2.1 Influence of noise on the amplitude estimation

Uncertainty of the component amplitude estimation mainly depends on the uncertainties of the amplitude DFT coefficients. Equation (19) is valid for all amplitude coefficients of the DFT that are large enough and sufficiently (half of the main lobe width) moved away from the margins of the spectral field ($\theta = 0,\ N/2$).

The price for the effective leakage reduction is in the increase of the estimation uncertainties, related to the unbiased CRB fixed by the signal-to-noise-ratio for a particular component. In Fig. 8, there are standard uncertainties of the amplitude estimation related to the CRB (33) (Petri, 2002) for the three-point estimation.

$$\sigma_{A_m} \geq \frac{1}{\sqrt{SNR}} \frac{1}{\sqrt{N}} = \sigma_{\mathrm{CRB},A} \tag{33}$$

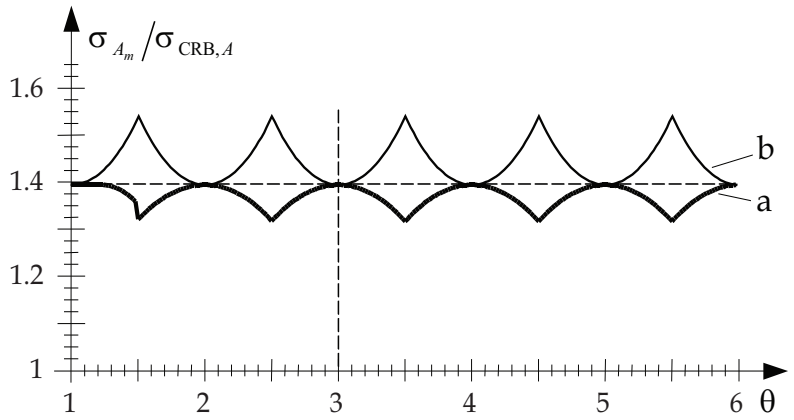


Fig. 8. Standard uncertainty of the amplitude three-point estimation using the Hann window related to the CRB (33) (a: θ is estimated, b: θ is known)

The distortions of the DFT coefficients and the number of points in the interpolation, have significant influence on the uncertainty of displacement δ_m, and amplitude A_m, succesively. As with the frequency estimation, the systematic errors decrease when increasing the number of points. Increasing the number of DFT coefficients used in the interpolation is reasonable, until the systematic error drops under the noise error (Fig. 9). After this point, by increasing the relative frequency θ_m (the number of periods of the measured signal in the measurement interval), or increasing spacing between two frequency components ($\propto 2 \cdot \theta_m$), the number of interpolation points can be decreased. A smaller number of the DFT coefficients in the calculation produces lower noise distortion, which becomes dominant in the final result.

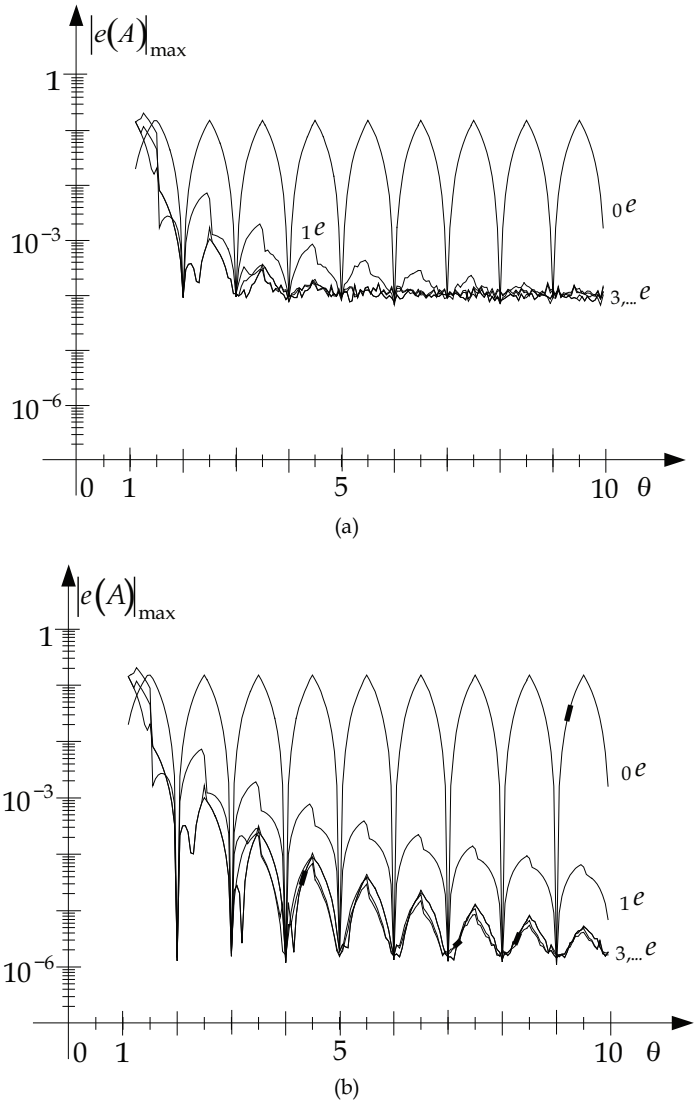


Fig. 9. The influence of the quantization noise on the amplitude estimation with 10-bit A/D converter (a) and 16-bit A/D converter (b)

2.3 Phase estimation

The second parameter of the signal component, besides the amplitude of the frequency main lobe, is the phase, i.e. the time position of the signal structure. As with previous estimations, the function $W(\theta)$ has to be analytically known. For the rectangular window with a large number of points $N >> 1$, the following equation is valid, where the Dirichlet kernel is used (Harris, 1978):

$$W_{\text{rect.}}(\theta)=\frac{\sin(\pi\theta)}{N\sin(\pi\theta/N)}\cdot e^{-j\pi\left(\frac{N-1}{N}\right)\theta} \tag{34}$$

The largest DFT coefficient, which is mostly composed of the short-range leakage contribution of the investigated component m, can be deduced from (5), and (34) using $a=\pi(N-1)/N$ and $-j=e^{-j\pi/2}$:

$$G(i_m)=\frac{A_m}{2}\left(\frac{\sin(\pi(i_m-\theta_m))}{N\sin(\pi(i_m-\theta_m)/N)}e^{j\left(a(\theta_m-i_m)+\varphi_m-\frac{\pi}{2}\right)}-\frac{\sin(\pi(i_m+\theta_m))}{N\sin(\pi(i_m+\theta_m)/N)}e^{-j\left(a(\theta_m+i_m)+\varphi_m+\frac{\pi}{2}\right)}\right) \tag{35}$$

The component phase φ_m is referred to as the start point of the window (not the middle point as is the case with the frequency and amplitude).

As N is usually large $N>>1$, and considering (6), equation (35) can be rewritten:

$$G(i_m)=\frac{A_m}{2}\left(\frac{\sin(\pi\delta_m)}{\pi\delta_m}e^{j\left(a(\delta_m)+\varphi_m-\frac{\pi}{2}\right)}-\frac{\sin(\pi\delta_m)}{\pi(2i_m+\delta_m)}e^{-j\left(a(2i_m+\delta_m)+\varphi_m+\frac{\pi}{2}\right)}\right) \tag{36}$$

Both, amplitude and phase have additional disturbing components from the second part in (36):

$$G(i_m)=|G(i_m)|e^{j\left(a(\delta_m)+\varphi_m-\frac{\pi}{2}\right)}\pm\Delta(i_m),\quad \arg(G(i_m))=\varphi_m+a\delta_m-\frac{\pi}{2}\pm\Delta\varphi(i_m) \tag{37}$$

If the displacement term is positive $0.5>\delta_m\geq 0$, then the second largest DFT coefficient is $G(i_m+1)$, and if the displacement term is negative $0>\delta_m\geq -0.5$, then the second largest DFT coefficient is $G(i_m-1)$ (Fig. 10). The largest side coefficient may commonly be expressed as:

$$G(i_m+s)=\frac{A_m}{2}\left(\frac{\sin(\pi(s-\delta_m))}{\pi(s-\delta_m)}e^{j\left(a(\delta_m-s)+\varphi_m-\frac{\pi}{2}\right)}-\frac{\sin(\pi(2i_m+s+\delta_m))}{\pi(2i_m+s+\delta_m)}e^{-j\left(a(\delta_m-s)+\varphi_m-\frac{\pi}{2}\right)}\right) \tag{38}$$

$$G(i_m+s)=|G(i_m+s)|e^{j\left(a(\delta_m-s)+\varphi_m-\frac{\pi}{2}\right)}\mp\Delta(i_m+s)$$

$$\arg(G(i_m+s))=\varphi_m+a(\delta_m-s)-\frac{\pi}{2}\mp\Delta\varphi(i_m+s) \tag{39}$$

Fig. 10, with large values of displacements, shows the amplitude and the phase differences between the phasors of the short-range contributions (dotted lines) and the complete phasors $G(i_m)$ and $G(i_m+s)$. It should be noted that differences have opposite signs at the largest two DFT coefficients surrounding the investigated component.

As we know, the spectrum with the Hann window can be obtained by shifting and weighing summations of the rectangular window spectrum (12). Using (34) in (12), it can be written:

$$W_{\text{H}}(\theta)=\frac{\sin(\pi\theta)}{2}\cdot e^{-ja\theta}\cdot\left(\frac{1}{N\sin(\pi\theta/N)}-\frac{1}{2}\left(\frac{-1}{N\sin(\pi(\theta+1)/N)}\cdot e^{-ja}+\frac{-1}{N\sin(\pi(\theta-1)/N)}\cdot e^{ja}\right)\right)$$

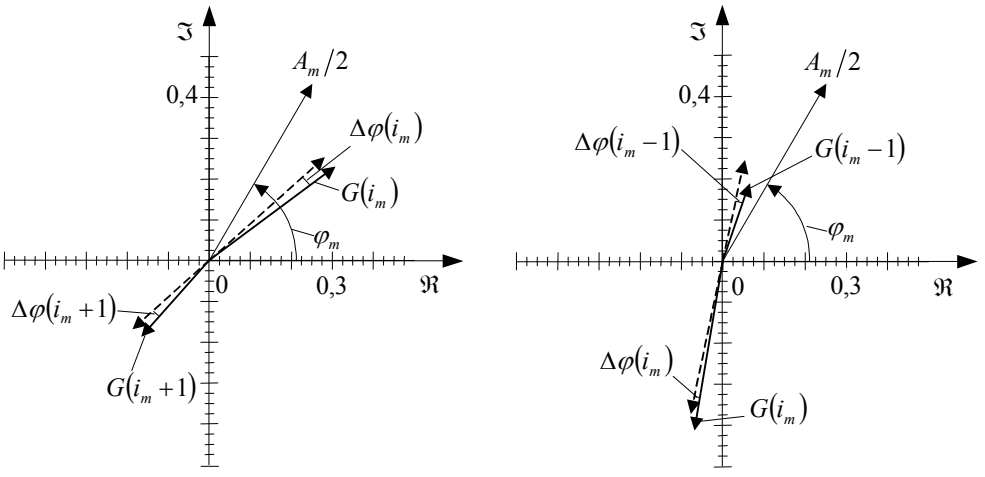


Fig. 10. Phasors diagrams for a single component $A_m = 1$, $\varphi_m = \pi/3$, and the rectangular window: a) $\theta_m = 2.4$, $i_m = 2$, $\delta_m = 0.4$; b) $\theta_m = 1.6$, $i_m = 2$, $\delta_m = -0.4$

If we have a lot of points in the measurement set $N >> 1$, the sine function can be approximated by $\sin(\pi\theta/N) \approx \pi\theta/N$. Considering also $e^{ja} \approx -1$ and $e^{-ja} \approx -1$, the expression in brackets can be simplified $\left(\frac{1}{\pi\theta} - \frac{1}{2}\left(\frac{1}{\pi(\theta+1)} + \frac{1}{\pi(\theta-1)}\right)\right) = \frac{1}{\pi\theta\left(1-\theta^2\right)}$, and we finally get:

$$W_{\mathrm{H}}(\theta) = \frac{1}{2}\frac{\sin(\pi\theta)}{\pi\theta\left(1-\theta^2\right)} \cdot e^{-ja\theta} \tag{40}$$

The largest DFT coefficient can be deduced from (5) and (40) considering $i_m - \theta_m = -\delta_m$ and $i_m + \theta_m = 2i_m + \delta_m$:

$$G_{\mathrm{H}}(i_m) = \frac{A_m}{4}\left(\frac{\sin(\pi\delta_m)}{\pi\delta_m\left(1-\delta_m^2\right)} e^{j\left(a(\delta_m)+\varphi_m-\frac{\pi}{2}\right)} - \frac{\sin(\pi\delta_m)}{\pi(2i_m+\delta_m)\left(1-(2i_m+\delta_m)^2\right)} e^{-j\left(a(2i_m+\delta_m)+\varphi_m+\frac{\pi}{2}\right)}\right) \tag{41}$$

As with the rectangular window, the second part in (41) causes additional disturbing components:

$$G_{\mathrm{H}}(i_m) = \left|G_{\mathrm{H}}(i_m)\right| e^{j\left(a(\delta_m)+\varphi_m-\frac{\pi}{2}\right)} \pm \Delta(i_m), \quad \left|G_{\mathrm{H}}(i_m)\right| = \frac{A_m}{4}\frac{\sin(\pi\delta_m)}{\pi\delta_m\left(1-\delta_m^2\right)}$$

$$\arg\left(G_{\mathrm{H}}(i_m)\right) = \varphi_{i_m} = \varphi_m + a\delta_m - \frac{\pi}{2} \pm \Delta\varphi(i_m) \tag{42}$$

The largest side coefficient can be expressed in short form as:

$$G_{\mathrm{H}}(i_m+s)=\left|G_H(i_m+s)\right|\mathrm{e}^{\mathrm{j}\left(a(\delta_m-s)+\varphi_m-\frac{\pi}{2}\right)}\mp\Delta(i_m+s);\ \left|G_{\mathrm{H}}(i_m+s)\right|=\frac{A_m}{4}\frac{\sin\left(\pi(s-\delta_m)\right)}{\pi(s-\delta_m)\left(1-(s-\delta_m)^2\right)}$$

$$\arg\left(G_{\mathrm{H}}(i_m+s)\right)=\varphi_{i_m+s}=\varphi_m+a(\delta_m-s)-\frac{\pi}{2}\mp\Delta\varphi(i_m+s) \tag{43}$$

2.3.1 Reduction of the systematic error

In the first approximation using the rectangular window, the second term in (36) and (38) can be neglected and the phase of component can be estimated by:

$$\varphi_{m,\mathrm{R}}^{\mathrm{I}}=\arg\left[G(i_m)\right]-a\delta_m+\frac{\pi}{2} \tag{44}$$

$$\varphi_{m,\mathrm{R}}^{\mathrm{II}}=\arg\left[G(i_m+s)\right]+a(s-\delta_m)+\frac{\pi}{2} \tag{45}$$

Another possibility is to estimate the component phase only by the phase DFT coefficient itself, where φ_m is referred to the middle point of the measurement window. However, this method has the same weak point as (44) and (45), since it doesn't consider the long-range contributions of the window (Fig. 14d).

We can improve the estimation by considering the long-range contributions. Because the disturbing angle components $\Delta\varphi(*)$ in (37) and (39) are small, they can be exchanged by sine functions and approximated by quotients:

$$\frac{\Delta\varphi(i_m)}{\Delta\varphi(i_m+s)}\cong\frac{\sin\left[\Delta\varphi(i_m)\right]}{\sin\left[\Delta\varphi(i_m+s)\right]}=\frac{\left|\Delta(i_m)\right|}{\left|G(i_m)\right|}\frac{\left|G(i_m+s)\right|}{\left|\Delta(i_m+s)\right|} \tag{46}$$

Here, the maximal amplitude values $|*|$ from (36) and (38) are taken in approximation:

$$\frac{\left|\Delta(i_m)\right|}{\left|\Delta(i_m+s)\right|}\cdot\frac{\left|G(i_m+s)\right|}{\left|G(i_m)\right|}=\frac{2i_m+s+\delta_m}{2i_m+\delta_m}\cdot\frac{\left|G(i_m+s)\right|}{\left|G(i_m)\right|} \tag{47}$$

If considering only one component with the DFT coefficient index large enough $i_m>>1$, or we want to symmetrically 'equalize' the long-range leakage contributions coming from both sides of the frequency axis, considering a multi-component signal - non-parametric approach, leakages can be equalized $\left|\Delta(i_m)\right|\approx\left|\Delta(i_m+s)\right|$ and (46) rewritten as:

$$\frac{\Delta\varphi(i_m)}{\Delta\varphi(i_m+s)}\cong\frac{\left|G(i_m+s)\right|}{\left|G(i_m)\right|}=\frac{\delta_m}{s-\delta_m};\qquad(s-\delta_m)\Delta\varphi(i_m)\cong\delta_m\Delta\varphi(i_m+s) \tag{48}$$

The multiplication of (37) and (39) by the correction (48), and subsequent summation gives us an estimation of the phase, as an averaging of the two arguments $\varphi_{i_m}=\arg\left[G(i_m)\right]$ and $\varphi_{i_m+s}=\arg\left[G(i_m+s)\right]$ surrounding the component (Fig. 12b):

$$\varphi_{m,\mathrm{R}}^{\mathrm{III}} = \left(1-\left|\delta_m\right|\right)\cdot\varphi_{i_m} + \left|\delta_m\right|\cdot\varphi_{i_m+\mathrm{s}} + \frac{\pi}{2} \tag{49}$$

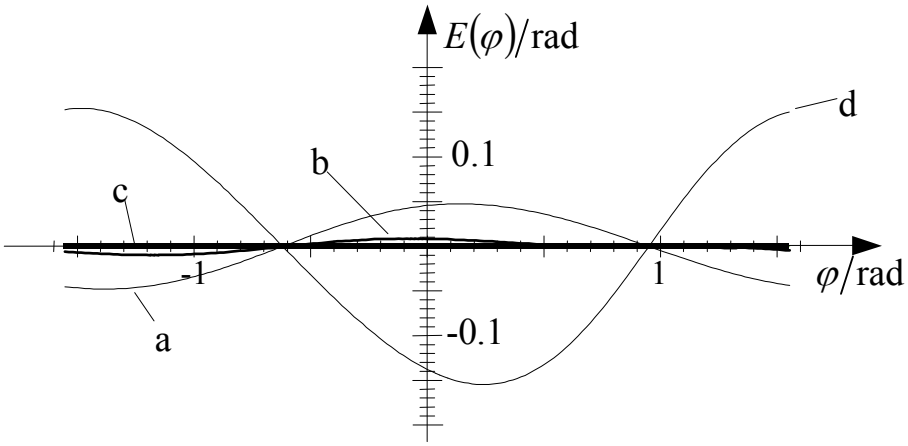


Fig. 11. Phase dependency errors at $\theta = 2.2$, $-\pi/2 \le \varphi \le \pi/2$; Estimations: a – by (44), b – by (49), c – by (51), d – by (45)

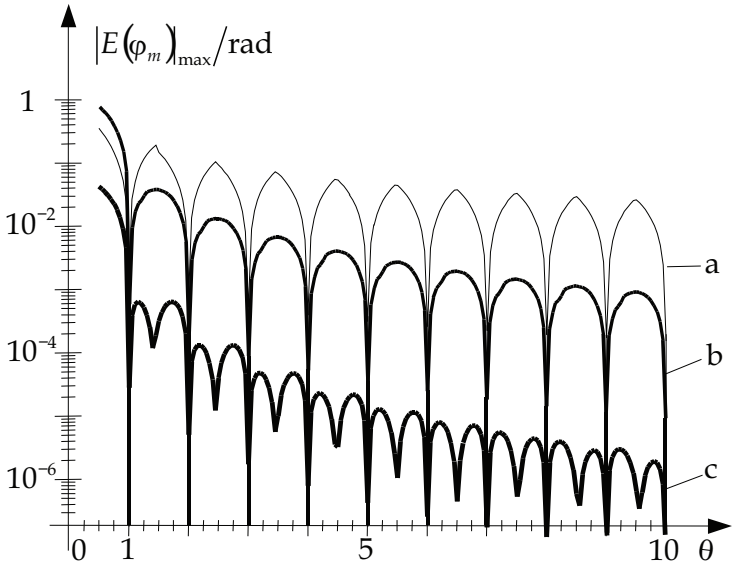


Fig. 12. Maximal systematic errors of the phase estimations with the rectangular window: a – by (44), b – by (49), c – by (51); θ is known

A better estimation can be obtained by also considering the long-range contributions in (47), when only one component is dominant in the signal – a more parametric approach (Fig. 12c):

$$\frac{\Delta\varphi(i_m)}{\Delta\varphi(i_m+s)} \cong \frac{2i_m+s+\delta_m}{2i_m+\delta_m}\cdot\frac{\left|G(i_m+s)\right|}{\left|G(i_m)\right|} = b\cdot\frac{\left|G(i_m+s)\right|}{\left|G(i_m)\right|} \tag{50}$$

$$\varphi^{\mathrm{IV}}_{m,\mathrm{R}} = \frac{\left|G(i_m)\right|\varphi_{i_m} + b\left|G(i_m+s)\right|\varphi_{i_m+s}}{\left|G(i_m)\right| + b\left|G(i_m+s)\right|} + sa\left(\frac{b\left|G(i_m+s)\right|}{\left|G(i_m)\right| + b\left|G(i_m+s)\right|} - \left|\delta_m\right|\right) + \frac{\pi}{2} \tag{51}$$

The systematic errors of the phase estimations $E=\varphi_m-\varphi_0$ (φ_0 - is the true value of the phase) are phase dependent (Fig. 11: The error curves are very close to the sine like functions). In simulations, the absolute maximum values of the errors at a given relative frequency have been searched when phase has been changed in intervals $-\pi/2\le\varphi\le\pi/2$ (Fig. 12). The estimation errors drop with the increasing relative frequency.

Fig. 13 shows the importance of the accuracy of the frequency estimation. If the frequency is estimated by a known two-point estimation (9) the overall errors increase (Fig. 3: $E_{\mathrm{c}^*}/E_{\mathrm{c}}\approx 200$).

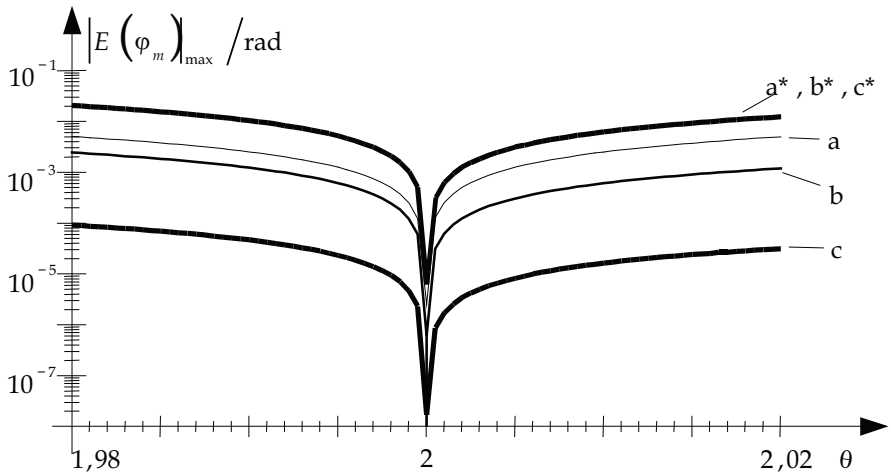


Fig. 13. Maximal systematic errors of the phase estimations with the rectangular window in the interval $1.98\le\theta\le 2.02$: a – by (44), b –by (49), c –by (51); a^*, b^*, c^* - θ is estimated by (9)

Using the Hann window, the expressions for phase have the same forms as for the rectangular window when the second term in (41) and (43) is neglected:

$$\varphi^{\mathrm{I}}_{m,\mathrm{H}} = \varphi_{i_m} - a\delta_m + \frac{\pi}{2} \tag{52}$$

$$\varphi^{\mathrm{II}}_{m,\mathrm{H}} = \varphi_{i_m+s} + a\left(s-\delta_m\right) + \frac{\pi}{2} \tag{53}$$

We can again improve the estimation by considering the long-range contributions, which have the following properties:

$$\frac{\Delta\varphi(i_m)}{\Delta\varphi(i_m+s)} \cong \frac{\sin\left(\Delta\varphi(i_m)\right)}{\sin\left(\Delta\varphi(i_m+s)\right)} = \frac{\left|\Delta(i_m)\right|}{\left|G_{\mathrm{H}}(i_m)\right|}\frac{\left|G_{\mathrm{H}}(i_m+s)\right|}{\left|\Delta(i_m+s)\right|} =$$

$$= \frac{(2i_m+s+\delta_m)\left(1-(2i_m+s+\delta_m)^2\right)}{(2i_m+\delta_m)\left(1-(2i_m+\delta_m)^2\right)} \cdot \frac{\left|G_{\mathrm{H}}(i_m+s)\right|}{\left|G_{\mathrm{H}}(i_m)\right|} \tag{54}$$

In the non-parametric approach, one can also equalize $\left|\Delta(i_m)\right| \approx \left|\Delta(i_m+s)\right|$ and (54) can be rewritten as:

$$\frac{\Delta\varphi(i_m)}{\Delta\varphi(i_m+s)} \cong \frac{\left|G_{\mathrm{H}}(i_m+s)\right|}{\left|G_{\mathrm{H}}(i_m)\right|} = \frac{\delta_m\left(1-\delta_m^2\right)}{(s-\delta_m)\left(1-(s-\delta_m)^2\right)} = \frac{1+s\delta_m}{2-s\delta_m} \tag{55}$$

If we equalize $(2-s\delta_m)\Delta\varphi(i_m) \cong (1+s\delta_m)\Delta\varphi(i_m+s)$, the equations (42) and (43) can be multiplied by corrections and added:

$$\varphi_{m,\mathrm{H}}^{\mathrm{I}} = \varphi_{i_m} - a\delta_m + \frac{\pi}{2} \pm \Delta\varphi(i_m) \quad \left|\cdot(2-s\delta_m)\right.$$

$$\varphi_{m,\mathrm{H}}^{\mathrm{II}} = \varphi_{i_m+s} + a(s-\delta_m) + \frac{\pi}{2} \mp \Delta\varphi(i_m+s) \quad \left|\cdot(1+s\delta_m)\right.$$

$$\varphi_{m,\mathrm{H}}^{\mathrm{III}} = \frac{(2-s\delta_m)\varphi_{i_m} + (1+s\delta_m)\varphi_{i_m+s}}{3} + \frac{a}{3}(s-2\delta_m) + \frac{\pi}{2} \tag{56}$$

When the sign of the displacement is positive $s=1$, the phase estimation (56) is made with arguments φ_{i_m} and φ_{i_m+1}:

$$\varphi_{m,\mathrm{H}}^{\mathrm{III}}\left(\varphi_{i_m},\varphi_{i_m+1}\right) = \frac{(2-\delta_m)\varphi_{i_m} + (1+\delta_m)\varphi_{i_m+1}}{3} + \frac{a}{3}(1-2\delta_m) + \frac{\pi}{2}, \tag{57}$$

and when the sign is negative $s=-1$, the estimation is done with φ_{i_m} and φ_{i_m-1}:

$$\varphi_{m,\mathrm{H}}^{\mathrm{III}}\left(\varphi_{i_m},\varphi_{i_m-1}\right) = \frac{(2+\delta_m)\varphi_{i_m} + (1-\delta_m)\varphi_{i_m-1}}{3} - \frac{a}{3}(1+2\delta_m) + \frac{\pi}{2} \tag{58}$$

The phase estimation can be improved further with averaging of the estimations by (57) and (58). With this averaging we get the three-point estimation (Fig. 14c):

$$\varphi_{m,\mathrm{H}}^{\mathrm{IV}}\left(\varphi_{i_m},\varphi_{i_m+1},\varphi_{i_m-1}\right) = \frac{\varphi_{m,\mathrm{H}}^{\mathrm{III}}\left(\phi_{i_m},\varphi_{i_m+1}\right) + \varphi_{m,\mathrm{H}}^{\mathrm{III}}\left(\varphi_{i_m},\varphi_{i_m-1}\right)}{2} \tag{59}$$

$$\varphi_{m,\mathrm{H}}^{\mathrm{IV}} = \frac{(1-\delta_m)\varphi_{i_m-1} + 4\varphi_{i_m} + (1+\delta_m)\varphi_{i_m+1}}{6} - \frac{2a\delta_m}{3} + \frac{\pi}{2} \tag{60}$$

The best results are obtained from the three-point estimation using the Hann window, when the frequency is known (Fig. 14c: $E(\varphi)_{\max} \le 1.7 \cdot 10^{-5}\,\mathrm{rad} \approx 1\,\mathrm{m}° \Leftarrow \theta > 5.5$). If the frequency has to be estimated, the overall error increases, but it is still under the error level of the one-point estimation (Fig. 14: $\mathrm{b}^* < \mathrm{a} \Leftarrow \theta > 5.5$). In Fig. 14, for error curve b^*, the frequency is estimated by the three-point interpolation (11).

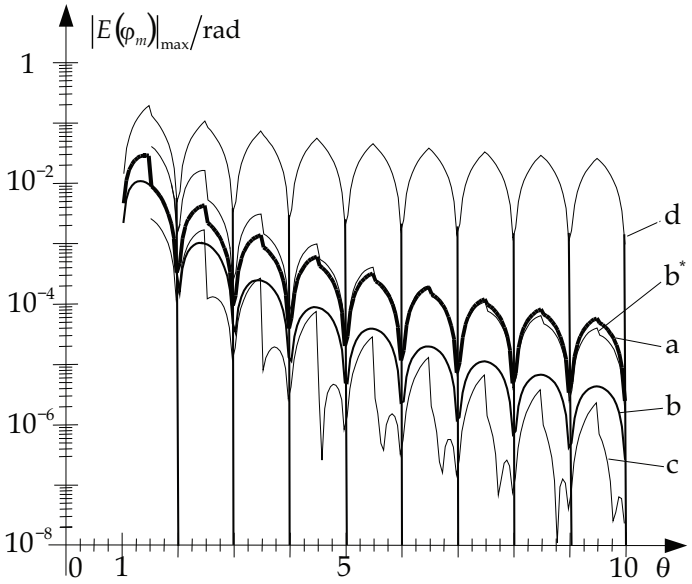


Fig. 14. Maximal systematic errors of the phase estimations with Hann window: a – one-point by (52), b – two-point by (56), c – three-point by (60), θ is known; b^* - θ is estimated by (11); and estimations using only phase DFT coefficient: d – the rectangular window, a – the Hann window

2.3.2 Uncertainty of the phase estimation

The uncertainty propagation through the DFT procedure is well known $\sigma_R = \sigma_I = \sigma_{|G(i)|} = \sigma_{\mathrm{DFT}} = \sigma_t \Big/ \left(N\sqrt{2}\right) \sqrt{\sum_{k=0}^{N-1} w^2(k)}$ (Agrež, 2007), where we use $R(i) = \mathrm{Re}\left[G(i)\right]$ and $I(i) = \mathrm{Im}\left[G(i)\right]$ for the real and imaginary parts of the DFT, respectively, and $|G(i)| = \sqrt{R^2(i) + I^2(i)}$ for the amplitude, and $\varphi(i) = \arg\left[G(i)\right] = \tan^{-1}\left(I(i)/R(i)\right)$ for the phase. The phase uncertainty is equal to the uncertainty of the DFT procedure scaled by the amplitude coefficient $\sigma_{\varphi(i)} = \sigma_{\mathrm{DFT}} \big/ |G(i)|$:

$$c_R(i) = \frac{\partial \varphi(i)}{\partial R(i)} = -\frac{I(i)}{|G(i)|^2}; \quad c_I(i) = \frac{\partial \varphi(i)}{\partial I(i)} = \frac{R(i)}{|G(i)|^2} \tag{61}$$

$$\sigma_{\varphi(i)}^2 = \left(c_R\sigma_R\right)^2 + \left(c_I\sigma_I\right)^2 = \sigma_{\mathrm{DFT}}^2\frac{I^2+R^2}{\left|G(i)\right|^4} = \frac{\sigma_{\mathrm{DFT}}^2}{\left|G(i)\right|^2} \tag{62}$$

It is evident that the standard uncertainty of the phase depends on the amplitude of the component. Moreover, in non-coherent sampling it changes with displacement δ as we see in the following examples.

First, we consider that frequency is known ($\sigma_\delta = 0$). For ease of understanding, one can omit index m in (44) $\varphi_{\mathrm{a}} = \varphi(i) - a\cdot\delta + \pi/2$, and from (62) we get (Fig. 15a):

$$\frac{\sigma_{\varphi_{\mathrm{a}}}}{\sigma_{\mathrm{DFT}}} = \frac{1}{\left|G(i)\right|} \tag{63}$$

For the second estimation by $\varphi_{\mathrm{b}} = (1-s\delta)\varphi(\mathrm{i}) + s\delta\,\varphi(\mathrm{i}+\mathrm{s}) + \pi/2$ (49) one needs sensitivity coefficients associated with the real and imaginary coefficients for two spectral lines i , $i+s$:

$$c_{R_{\mathrm{b}},i} = \frac{\partial\varphi_{\mathrm{b}}(i)}{\partial R(i)} = (1-s\delta)c_R(i);\quad c_{I_{\mathrm{b}},i} = \frac{\partial\varphi_{\mathrm{b}}(i)}{\partial I(i)} = (1-s\delta)c_I(i)$$

$$c_{R_{\mathrm{b}},i+s} = \frac{\partial\varphi_{\mathrm{b}}(i+s)}{\partial R(i+s)} = s\delta\,c_R(i+s);\quad c_{I_{\mathrm{b}},i+s} = \frac{\partial\varphi_{\mathrm{b}}(i+s)}{\partial I(i+s)} = s\delta\,c_I(i+s) \tag{64}$$

As the correlation coefficients for the rectangular window $r\left(R(i),R(i+s)\right)$ and $r\left(I(i),I(i+s)\right)$ are zero, and standard uncertainties are equal $\sigma_{R,*} = \sigma_{I,*} = \sigma_{\mathrm{DFT}}$ $(*=i,i+s)$ we can write according to (JCGM, 2008) (Fig. 15b):

$$\left(\frac{\sigma_{\varphi_{\mathrm{b}}}}{\sigma_{\mathrm{DFT}}}\right)^2 = c_{R_{\mathrm{b}},i}^2 + c_{I_{\mathrm{b}},i}^2 + c_{R_{\mathrm{b}},i+s}^2 + c_{I_{\mathrm{b}},i+s}^2;\quad \frac{\sigma_{\varphi_{\mathrm{b}}}}{\sigma_{\mathrm{DFT}}} = \sqrt{\left((1-s\delta)/\left|G(i)\right|\right)^2 + \left(\delta/\left|G(i+s)\right|\right)^2} \tag{65}$$

In calculations of the four sensitivity coefficients for the third estimation (51), one needs partial sensitivity coefficients for the amplitude $\partial\left|G(*)\right|/\partial R(*) = R(*)/\left|G(*)\right|$, $\partial\left|G(*)\right|/\partial I(*) = I(*)/\left|G(*)\right|$ and for the phase coefficients $\partial\varphi(*)/\partial R(*)$, $\partial\varphi(*)/\partial I(*)$ (61), since all of them contribute in the estimation:

$$\varphi_{\mathrm{c}} = \frac{\left|G(i)\right|\varphi(\mathrm{i}) + b\left|G(i+s)\right|\varphi(\mathrm{i}+\mathrm{s})}{\left|G(i)\right| + b\left|G(i+s)\right|} + sa\left(\frac{b\left|G(i+s)\right|}{\left|G(i)\right| + b\left|G(i+s)\right|} - s\delta\right) + \frac{\pi}{2} \tag{66}$$

$$c_{R_{\mathrm{c}},*} = \frac{\partial\varphi_{\mathrm{c}}(*)}{\partial R(*)};\qquad c_{I_{\mathrm{c}},*} = \frac{\partial\varphi_{\mathrm{c}}(*)}{\partial I(*)} \tag{67}$$

In this case the uncertainty is close to the uncertainty of the estimation by (49) (Fig. 15c).

The price for the effective leakage reduction is in the increase of the estimation uncertainties related to the unbiased CRB. Ratios of the uncertainties of the phase estimations related to the CRB (68), are between 1 and 1.7 symmetrically, depending upon the term δ at higher values of the relative frequency θ (Fig. 15).

$$\sigma_{\varphi_m} \geq 2\frac{1}{\sqrt{SNR}}\frac{1}{\sqrt{N}} = \sigma_{\mathrm{CRB},\varphi} \tag{68}$$

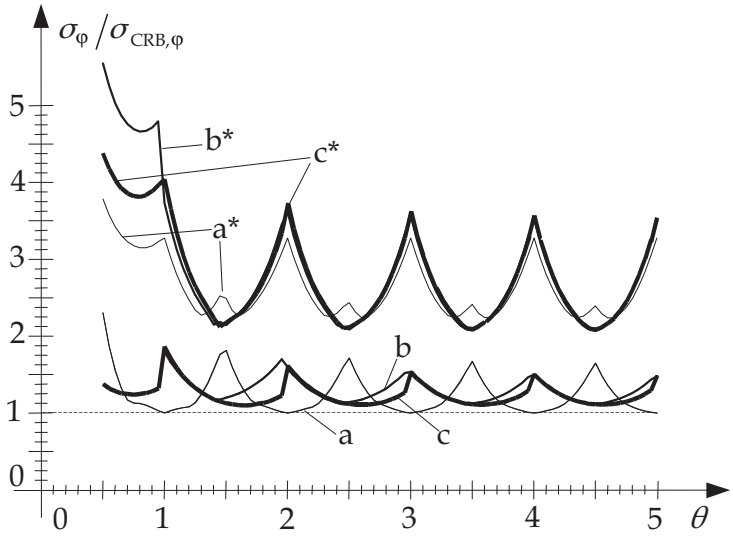


Fig. 15. Ratios of the uncertainties of the phase estimations with rectangular window related to the CRB (68). Estimations: a – by (44), b – by (49), c – by (51), θ is known; a^*, b^*, c^* - θ is estimated by (9b)

The uncertainties of the estimations increase if one needs to also estimate the displacement term $\delta\left(|G(i)|, |G(i+s)|\right)$. In estimation algorithms (44), (49), and (51), one also needs partial sensitivity coefficients for the displacement terms $\partial\delta/\partial|G(i)|$ and $\partial\delta/\partial|G(i+s)|$. Fig. 15 shows that uncertainties of the estimations increase for a factor $2 \div 2.3$ if frequency is estimated by (9b). The uncertainties of the estimations where frequency has to be estimated first, are very close to each other (Fig. 15: $E_{a^*} \approx E_{b^*} \approx E_{c^*}$) at higher values of θ. For the Hann window, the uncertainty levels increase (62), as the main amplitude coefficient decreases by a factor of 2 with respect to the rectangular window.

3. Conclusion

In chapter, the non-parametric interpolated DFT algorithms for emulating coherent sampling are described. The advantages of the DFT interpolations for the frequency, amplitude, and phase of the signal component are identified. Interpolations where the long-range leakage is considered, illustrate a decrease in systematic effects. The algorithms retain all benefits of the DFT approach and improve the estimation accuracy adaptively for a

particular component as a function of its frequency position. The weights emphasize the DFT coefficients of the spectrum peak related to the investigated component. The spectrum of the window used must be formally well-known, like the Hann window, for a better analytical expression. Interpolation with a larger number of the DFT coefficients decreases the systematic errors. It can be concluded, that if we selectively use a different number of the DFT coefficients in the interpolation algorithms for a particular component of the signal, we adapt the apparent window shape for that component. A trade-off between a reduction in systematic error of the parameter estimation and the uncertainty of the estimated results is highlighted. The use of a suitable interpolation algorithm depends on the level of the noise floor of the acquisition channel, or better *SNR*, and on the position of the frequency component along the frequency axis.

4. References

Agrež D. (2002). Weighted Multi-Point Interpolated DFT to Improve Amplitude Estimation of Multi-Frequency Signal. *IEEE Transactions on Instrumentation and Measurement*, Vol. 51, No. 2, April 2002, pp. 287-292, ISSN 0018-9456

Agrež, D. (2007). Dynamics of frequency estimation in the frequency domain. *IEEE Transactions on Instrumentation and Measurement*, Vol. 56, No. 6, December 2007, pp. 2111-2118, ISSN 0018-9456

Belega, D. & Dallet, D. (2009). Multifrequency signal analysis by interpolated DFT method with maximum sidelobe decay windows. *Measurement*, Vol. 42, No. 3, 2009, pp. 420-426, April 2009, ISSN 0263-2241

D'Antona G. & Ferrero, A. (2006). *Digital Signal Processing for Measurement Systems. Theory and Applications*, Springer Science 2006, ISBN-10: 0-387-24966-4

Gabor, D. (1946). Theory of communication. *Journal of the IEEE*, Vol. 93, pp. 429-457

Harris, F. J. (1978). On the use of windows for harmonic analysis with the discrete Fourier transform. *Proceedings of the IEEE*, Vol. 66, No. 1, January 1978, pp. 51-83, ISSN 0018-9219

JCGM (2008). *Evaluation of measurement data – Guide to the expression of uncertainty in measurement*. Joint Committee for Guides in Metrology [JCGM] 100:2008, First edition, September 2008

Novotný, M. & Sedláček, M. (2010). The influence of window sidelobes on DFT-based multifrequency signal measurement. *Computer Standards & Interfaces*, Vol. 32, No. 3, March 2010, pp. 110-118, ISSN 0920-5489

Petri, D. (2002). Frequency-domain testing of waveform digitizers. *IEEE Transactions on Instrumentation and Measurement*, Vol. 51, No. 3, 2002, pp. 445-453, ISSN 0018-9456

Schoukens, J., Pintelon, R. & Van hamme, H. (1992). The interpolated fast Fourier transform: A comparative study. *IEEE Transactions on Instrumentation and Measurement*, Vol. 41, No. 2, April 1992, pp. 226-232, ISSN 0018-9456

Siebert, W. McC. (1986) *Circuits, Signals and Systems*, The MIT Press, ISBN 0-262-19229-2, McGraw-Hill, Cambridge, New York, ..., pp. 497-502

Solomon, O. M. (1992). The effects of windowing and quantization error on the amplitude of frequency-domain functions. *IEEE Transactions on Instrumentation and Measurement*, Vol. 41, No. 6, December 1992, pp. 932-937, ISSN 0018-9456

Widrow, B. & Kollar, I. (2008) *Quantization Noise*, Cambridge Univ. Press 2008, ISBN 978-0-521-88671-0, Cambridge, New York,

9

A Reformulative Retouch on the Fourier Transform – "Unprincipled" Uncertainty Principle

Csaba Szántay, Jr.
Gedeon Richter Plc.
Hungary

1. Introduction

According to a widespread notion, a monochromatic harmonic temporal wave's frequency ω becomes inherently "uncertain" to a degree $\Delta\omega$ if the wave lasts for only a limited time Δt owing to the "Uncertainty Principle" which states that $\Delta t \cdot \Delta\omega \geq$ constant (i.e. $\Delta\omega$ cannot be zero if Δt is finite). This frequency uncertainty is commonly thought of as also being reflected in the fact that the ("sinc"-shaped) frequency spectrum obtained by Fourier-transforming a finite-duration monochromatic wave spreads over a nonzero frequency range. Since, as often argued, the Fourier spectrum represents the "frequency components" of a temporal function (or signal), the nonzero width of the spectrum tells us that the "nominally" monochromatic ω frequency of a time-limited sinusoid is "effectively" polychromatic, with frequencies continuously distributed over a $\Delta\omega$ range in accord with the Uncertainty Principle.

A typical area where this argument shows up is Pulsed Fourier-Transform (PFT) nuclear magnetic resonance (NMR) spectroscopy. In NMR a molecular ensemble is placed in a strong homogeneous static magnetic field, whereupon the ensemble's NMR-active nuclei (those that act like spinning atomic magnets, say protons) will precess about the direction of the magnetic field. Those "spins" in the ensemble which precess (Larmor precession) with identical ω_L frequencies form a net macroscopic magnetization which is normally aligned along the external field, but which will also (Larmor-)precess with a frequency ω_L following a transient perturbation of the magnetization from its equilibrium state. Typically there are several such bundles of homogeneous spins in the analyzed sample, covering a range of Larmor frequencies $\Delta\omega_L$. PFT-NMR works on the principle that the sample's bulk magnetizations of interest can be excited into precession-mode (made to "resonate") simultaneously in a wide $\Delta\omega_L$ ("off-resonance") frequency range when perturbed by a "driving" electromagnetic wave of a single well-defined monochromatic radio-frequency (RF) ω_D if the wave is applied as a "pulse" in a sufficiently short time Δt. Because this phenomenon may appear counter-intuitive, the fundamental NMR literature is abundant in attempts to offer a preliminary explanation for why pulsed excitation works, typically invoking the "Uncertainty Principle" and the "sinc"-shaped FT-spectrum of the pulse according to the notions outlined above. Some representative examples from authoritative

sources are as follows: "…although the applied excitation may be precisely centred at a frequency ω_D , […], our act of turning the excitation power on at time zero and and off at time Δt effectively broadens the spectral range of the excitation (to a bandwidth of ~$1/\Delta t$)." (Marshall and Verdun, 1990); "A pulse of monochromatic RF with a rectangular envelope can be described in the frequency domain as a band of frequencies centred at the RF frequency. The Heisenberg principle states that there is a minimum uncertainty in the simultaneous specification of […] the frequency of a system and the duration of the measurement. […] this means that the […] irradiation is spread over a wide frequency band. […] the 'sinc' Fourier spectrum of a rectangular RF pulse shows that a shorter pulse gives a wider 'sinc' band and a longer pulse gives a narrower 'sinc' band." (King and Williams, 1989); "…if the pulse is made shorter, we will no longer have a truly monochromatic Fourier spectrum even though the source is still monochromatic. This is because many different frequencies have to be combined in order to form the rising and falling edges of the rectangular pulse." (Fukushima and Roeder, 1981); "… as the Uncertainty Principle indicates, a pulse [of carrier frequency ω_D] will contain, in effect, a range of frequencies centred on ω_D . […] the distribution of RF magnetic field amplitudes takes the 'sinc' form […] which is the frequency-domain equivalent of a short pulse in the time domain. The two domains are connected by the Fourier transform." (Harris, 1983); "… the RF source […] is monochromatic, so we have to work out a way of using a single frequency to excite multiple frequencies. To see how this can be done we take our cue from the Uncertainty Principle. If the irradiation is applied for a time Δt, then […] the nominally monochromatic irradiation is uncertain in frequency by about $1/\Delta t$." (Derome, 1987)

One should quickly note that at a macroscopic level the basic NMR phenomenon is a purely classical effect (Hoult, 1989 and Hanson, 2008) involving interactions between the oscillating magnetic component of the RF pulse and the sample's bulk magnetizations. Indeed, the fact that a monochromatic pulse can excite spins in a wide frequency range can be well explained classically using the laws of driven mechanical harmonic oscillators and magnetism (Bloch, 1946) without resorting to the vague idea that the ω_D RF pulse frequency should be "uncertain" or "effectively" polychromatic according to its Fourier spectrum. Furthermore, except for one case the above arguments are unclear on whether by "Uncertainty Principle" they mean the Heisenberg Uncertainty Principle (HUP) which is a probabilistic, quantum-mechanical statement, or the Fourier Uncertainty Principle (FUP) which is a deterministic theorem in the discipline of time-frequency analysis.

In a series of recent papers (Szántay, 2007, 2008a, 2008b, 2008c) the author of the present work explored the above problem in detail, showing that a) the idea that the frequency of a time-limited sinusoid would be uncertain in the Heisenberg sense is flawed; b) it is the FUP (and not the HUP) that is of relevance to the problem; c) the notion that a monochromatic pulse "effectively" comprises of a range of physically existing frequency components is erroneous. During that discourse the need had come up to contemplate the essence of the Fourier Transform (FT) itself, which may appear somewhat odd considering the fact that the FT is ubiquitously applied in a broad range of sciences and technology, and works explaining its mathematics and its manifold physical applications are available in almost countless abundance. However, it turns out that the FT is so commonplace, it has become so much consolidated into many fields of science and engineering, its basic principles and formulas are so commonly taken for granted, that, paradoxically, this can easily stop people

from thinking more deeply about how and why the FT works as a mathematical tool and what exactly it means when evoked to interpret physical phenomena. One of the central features of the present author's approach in discussing the FT was the use of a "touched up" mathematical formalism to provoke a new way of looking at some key ideas of the FT and to avoid some hidden ambiguities in meaning that are otherwise inherent to the conventional formalism and can give rise to the misconceptions mentioned above. Besides coming out with a disambiguative but symbolically elaborate formalism for the FT involving the depiction of the domain intervals on which functions "exist" and introducing the concept of thinking about the FT in either "trigonometric-frequency-space" or "phasor-frequency-space", those articles also expounded, from a new perspective, on several apparently simple but in practice elusive related concepts such as "periodicity", "harmonic oscillation", "phasor", "negative frequency", "uncertainty", etc.; much emphasis was placed also on a comparison of Fourier analysis and the Fourier transform, on the pros and cons of metaphoric thinking and the dangers of unrecognized linguistic ambiguities (e.g., words and phrases like "*uncertainty*", "*nominally* monochromatic" and "*effectively* polychromatic" deceptively appear to be meaningful, but on deeper examination their meaning is ambiguous and notoriously misunderstood).

Owing to the volume and thematic complexity of those papers the need has repeatedly arisen to present its main thrusts in a more compact and symbolically simplified form. The present work therefore attempts to condense the original discussion into a shorter communication by focusing only on two themes, namely the reformulation of the primary FT equations (other closely related topics such as Fourier analysis will be left out) and the main conclusions on the problem of the "Uncertainty Principle". These thoughts are hoped to provide some fresh insights and a deeper understanding of the essence of the FT and the Uncertainty Principles. In that regard it should be emphasized that the applied mathematical formalism is not a l'art pour l'art design: it serves to facilitate a more conscientious way of thinking about the pertinent topics. For the full technical and philosophical details behind the ideas presented here the reader is encouraged to consult the original articles.

2. Nomenclature

2.1 General notation

- The term *Fourier analysis* will be used to refer to the Fourier series expansion of periodic functions and the term *Fourier transform* (FT) will refer to the generalization of Fourier analysis including nonperiodic functions. The FT will be discussed only in the context of the Fourier-pair variables time (t) and frequency (ω). The word "spectrum" will in genereal mean the ω-dimension (ω-D) representation of a time-dimension (t-D) function, more specifically as it has been obtained by the FT.
- Sets will be represented by bold, italicized letters while vectors will be denoted by bold, non-italicized letters. If in a set $\boldsymbol{x}$ its members x can assume values on a continuous scale, I will use the symbol "@" as a primary generic index to tag an arbitrary value as "$x_@$" so as to emphasize that we are focusing on that particular value in the set ("@" can be interpreted as: "*at* a given value". Members x of the sets $\boldsymbol{x}$, $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$ can in general take on negative as well as positive values, only nonnegative values, and only nonpositive values, respectively. If a set $\boldsymbol{x}$ represents an interval for the variable

$x \in \mathbb{X}$, we will consider five main interval types labeled as follows: $\overset{\Leftrightarrow}{\mathbb{X}} = (x = -\infty, x = \infty)$ is open and unbounded on both sides; $\overset{\bullet\!-\!\bullet}{\mathbb{X}} = [x_@, x_{@'}]$ is closed and bounded (compact) at arbitrary $x_@$ and $x_{@'}$ values on both sides; $\overset{\Rightarrow}{\mathbb{X}} = [x = 0, x = \infty)$ is closed and bounded at zero on the left, open and unbounded on the right and $\overset{\Leftarrow}{\mathbb{X}} = (x = -\infty, x = 0]$ is closed and bounded at zero on the right and open and unbounded on the left; $\overset{\uparrow}{\mathbb{X}}_@ = \{x | x_@ \le x \le x_{@'}, x_@ \leftarrow x_{@'}\}$ will be associated with a function which has a finite function value at $x = \overset{\bullet}{x}_@$ but is zero if $x \ne \overset{\bullet}{x}_@$; $\overset{\uparrow}{\mathbb{X}}_@ = \{x | x_@ \le x \le x_{@'}, x_@ \leftarrow x_{@'}\}$ will be used in connection with a Dirac delta whose function value is infinite at $x = \overset{\uparrow}{x}_@$ but zero if $x \ne \overset{\uparrow}{x}_@$.

- Underlined symbols are used to denote complex quantities having a real ($\Re$) and an imaginary ($\Im$) component, so a complex constant $\underline{A} = ({}^{\Re}A, {}^{\Im}A)$ is defined and symbolized as $\underline{A} = {}^{\Re}A + i \cdot {}^{\Im}A = {}^{\Re}A + {}^{\Im}A'$ where $i \cdot {}^{\Im}A = {}^{\Im}A'$. Hence $\underline{A}$ has the absolute value $|\underline{A}| = A = \sqrt{({}^{\Re}A)^2 + ({}^{\Im}A)^2} = \sqrt{({}^{\Re}A^2) - ({}^{\Im}A')^2}$. A complex number $\underline{A}$ expressed in the form of a complex exponential $\underline{A} = A \cdot \exp(i \cdot \varphi)$ will be referred to as a *phasor* where the phase φ is measured from the positive real axis ($+\Re$) with a counterclockwise angle being positive and a clockwise angle being negative. A non-underlined character, if lacking the tags $\Re$ or $\Im$, represents a real-valued quantity by default. Likewise, a function in general is represented as $f(x)$ if it is real-valued and as $\underline{f}(x)$ if it is complex-valued.
- The action of a mathematical formula (rule, operator, etc…) upon a set will be denoted as $\text{Formula}\langle set \rangle$. Thus, a rule acting on a variable x to give a function $\underline{f}(x)$ is written as $\underline{f}(x) = \text{Rule}\langle x \rangle$, and an operator that converts a function $\underline{f}(x)$ into $\underline{F}(y)$ is depicted as $\underline{F}(y) = \text{Operator}\langle \underline{f}(x) \rangle$.
- The natural (largest possible) domain of a function $\underline{f}(x) = \text{Rule}\langle x \rangle$ is the set of all allowable inputs that the function's argument x may assume. However, $\underline{f}(x)$ can of course be restricted to or can simply be of interest to us over a subset $\mathbb{X}$ of the natural domain, and herein $\mathbb{X}$ will be called the *operative interval* of $\underline{f}(x)$. A function's operative interval will be denoted as $\underline{f}(x)\big|_{\mathbb{X}} = \text{Rule}\langle x \rangle\big|_{\mathbb{X}}$ and should be interpreted either as having no values or zero function values for any $x \notin \mathbb{X}$. For example, for a single rectangular "bump" along x we have $\mathbb{X} = \overset{\bullet\!-\!\bullet}{\mathbb{X}}$ corresponding to the rectangle's width, while for a function that starts with a finite value at $x = 0$ but decays exponentially to zero at $x = \infty$ we have $\mathbb{X} = \overset{\Rightarrow}{\mathbb{X}}$ ranging from zero to positive infinity. Care must be taken not to confuse the operative interval (which is always indicated as a subscript following a bar

placed after the function's name or rule of association) with the function's natural domain. For example, for a function $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}}}^{\bullet\text{–}\bullet}$ the natural domain may be $\overset{\Leftrightarrow}{\boldsymbol{\mathcal{X}}}$ but the operative interval is $\overset{\bullet\text{–}\bullet}{\boldsymbol{\mathcal{X}}} \in \overset{\Leftrightarrow}{\boldsymbol{\mathcal{X}}}$; for $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}}}^{\Leftrightarrow}$ the operative interval is $\overset{\Leftrightarrow}{\boldsymbol{\mathcal{X}}}$, implying that the natural domain must also be $\overset{\Leftrightarrow}{\boldsymbol{\mathcal{X}}}$. A proper understanding of the FT requires that one should always be conscious of what domains and intervals are relevant to the time scales and frequency ranges in the pertinent mathematical expressions. These aspects are often far from being self-evident from the formulas themselves in their conventional notation, and are usually present as hidden assumptions or unstated premises. To overcome this problem herein the pertinent interval types will be represented as $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}}}^{\Leftrightarrow} = \text{Rule}\langle x\rangle\big|_{\boldsymbol{\mathcal{X}}}^{\Leftrightarrow}$, $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}}}^{\bullet\text{–}\bullet} = \text{Rule}\langle x\rangle\big|_{\boldsymbol{\mathcal{X}}}^{\bullet\text{–}\bullet}$; $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}}}^{\Rightarrow} = \text{Rule}\langle x\rangle\big|_{\boldsymbol{\mathcal{X}}}^{\Rightarrow}$; $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}}}^{\Leftarrow} = \text{Rule}\langle x\rangle\big|_{\boldsymbol{\mathcal{X}}}^{\Leftarrow}$, $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}}_{@}}^{\uparrow} = \underline{A}_{@}\delta_{@}(x - x_{@}) = \underline{A}_{@} \cdot \infty$ if $x = x_{@}$, otherwise zero, and $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}}_{@}}^{\bullet} = \underline{A}_{@}\big|_{\boldsymbol{\mathcal{X}}_{@}}^{\bullet} = \underline{A}_{@}$ if $x = x_{@}$, otherwise zero. If the properties of $\underline{f}(x)$ are considered on different intervals simultaneously, the pertinent interval symbols will be separated by semicolons, such as $\underline{f}(x)\big|_{\boldsymbol{\mathcal{X}};\,\boldsymbol{\mathcal{X}}}^{\Leftrightarrow;\,\bullet\text{–}\bullet} = \text{Rule}\langle x\rangle\big|_{\boldsymbol{\mathcal{X}};\,\boldsymbol{\mathcal{X}}}^{\Leftrightarrow;\,\bullet\text{–}\bullet}$.

2.2 Trigonometric and phasor frequencies

The symbol ω will be used as a general representation of (angular) frequency such that the oscillation frequencies associated with a temporal function have constant values ($\omega(t) = \text{constant}$) over the operative interval $\boldsymbol{t}$ of the function. In the customary notation a sinusoid (trigonometric) harmonic wave with a complex amplitude (i.e., a 1D oscillation in the complex plane that gives a 2D wave as a function of time) has the general form $\underline{A}(\omega) \cdot \cos[(\omega t) + \varphi^{\circ}(\omega)]$ and a *harmonic phasor* (i.e., a vector **A** rotating uniformly either clockwise or anti-clockwise in the complex plane, and describing a 3D helix as a function of time) has the general form $A(\omega) \cdot \exp[i \cdot \varphi^{\circ}(\omega)] \cdot \exp(i \cdot \omega t) = \underline{A}(\omega) \cdot \exp(i \cdot \omega t)$, where φ° is the wave's initial phase at t = 0. For the harmonic phasor the phase angle $\varphi(t) = \omega t + \varphi^{\circ}$ is measured from the positive real axis (+ℜ) with a counterclockwise angle being positive and a clockwise angle being negative. For simplicity both a sinusoid wave and a harmonic phasor will be referred to as a "harmonic wave".

The sinusoid and phasor forms of a harmonic wave are connected by Euler's famous identity, written in its conventional form as $\exp(i \cdot \omega t) = \cos(\omega t) + i \cdot \sin(\omega t)$, from which the trigonometric terms can be expressed by rearrangement as $\cos(\omega t) = 0.5 \cdot \exp(i \cdot \omega t) + 0.5 \cdot \exp(-i \cdot \omega t)$, and $\sin(\omega t) = -i \cdot 0.5 \cdot \exp(i \cdot \omega t) + i \cdot 0.5 \cdot \exp(-i \cdot \omega t)$. Using this conventional notation the concept of the sign of ω, in particular the idea of a negative frequency, can be a confusing issue. Thus, based on making a distinction between "trigonometric frequency" and "phasor frequency" the following notation is introduced: a *trigonometric frequency* (the frequency of a sinusoid oscillation) will be labeled as $\varpi \in \boldsymbol{\varpi}$; a *phasor frequency* (the frequency of harmonic phasor) will be labeled as $\wp \in \boldsymbol{\wp}$. Accordingly, a sinusoid harmonic wave $\underline{h}_{\varpi}(t)$ and a harmonic phasor $\underline{h}_{\wp}(t)$ take the following general forms:

$$\underline{h}_{\varpi}(t) = \underline{A}(\varpi) \cdot \cos[(\varpi t) + \varphi^{\circ}(\varpi)] , \tag{1}$$

$$\underline{h}_{\phi}(t) = A(\phi) \cdot e^{i \cdot \varphi^{\circ}(\phi)} \cdot e^{i \cdot \phi t} = \underline{A}(\phi) \cdot e^{i \cdot \phi t} = {}^{\Re}A(\phi) \cdot e^{i \cdot \phi t} + i \cdot {}^{\Im}A(\phi) \cdot e^{i \cdot \phi t} . \tag{2}$$

Note from (2) that the phasor's complex amplitude $\underline{A}$ always corresponds to $t = 0$ since $\underline{A}$ is defined by its magnitude (absolute value) and the phasor's initial phase, which are both time-independent quantities. From (2) we see that the phasor $\underline{A}(\phi) \cdot \exp(i \cdot \phi t)$ can be equivalently viewed also as a sum of two phasors of the same frequency, one with a real amplitude ${}^{\Re}A$, and the other with a pure imaginary amplitude $i \cdot {}^{\Im}A$.

For the trigonometric wave $h_{\varpi}(t)$ the meaning of ϖ seems straightforward: it expresses the number of full periods (multiplied by 2π) completed by the wave in unit time, whereby ϖ is defined as having positive numerical values only since it would not make sense to speak of a negative number of oscillations per unit time. The algebraically positive nature of ϖ merits emphasis because in the context of the FT tigonometric frequencies are sometimes (mistakenly) given negative values in the literature [e.g. the fact that neither the cosine transform, nor the sine transform alone can distinguish between the two possible rotational directions of a harmonic phasor is often explained by the formal truths that $\cos[(-\omega)t] = \cos(\omega)t$ and $\sin[(-\omega)t] = -\sin(\omega)t$]. However, one should appreciate that the concept of $\cos[(-\omega)t]$ (or $\cos[(-\varpi)t]$) or $\sin[(-\omega)t]$ (or $\sin[(-\varpi)t]$) does not make any physical sense. This issue is partly rooted in the fact that the literature makes no symbolic distinction between "ϖ" and "ϕ", hence the intended meaning (phasor vs. trigonometric) of "frequency", labeled as "ω", is typically an implicit consequence of the context in which it appears but is not implied or conveyed by the symbol "ω" itself. Thus the formulas $\cos[(-\omega)t]$ or $\sin[(-\omega)t]$ may easily escape mental objection when speaking about positive and negative phasor frequencies when the latter are also labeled as $+\omega$ and $-\omega$. There is in fact no need to allow sinusoid frequencies to be negative (see below) and we can adhere to the physically appropriate definition that trigonometric frequencies are always positive.

A harmonic phasor $h_{\phi}(t)$ has either a positive or a negative frequency ϕ^{+} and ϕ^{-} depending on the direction of "skew" (or "helicity") of the 3D helix described by the phasor as a function of time. I employ the convention according to which for positive ϕ the phasor $\underline{h}_{\phi^{+}}(t) = A \cdot \exp(i \cdot \phi^{+} t)$ corresponds to the vector **A** rotating in the complex plane in the $+\Re \rightarrow +\Im \rightarrow -\Re \rightarrow -\Im$ direction as time progresses, while for negative ϕ the phasor $\underline{h}_{\phi^{-}}(t) = A \cdot \exp(i \cdot \phi^{-} t) = A \cdot \exp(-i \cdot \phi^{+} t)$ represents rotation in the opposite sense.

Note that according to the above considerations there is an intrinsic difference between the concepts of the "sign" of the frequency for a harmonic sinusoid wave and a harmonic phasor. If, in a $(\Re, \Im, t)$ space, we associate a positive skew with ϕ^{+} and a negative skew with ϕ^{-} (and this is what we mean by the positive and negative sign of ϕ), then a trigonometric frequency ϖ, which is "skew-less", should also be "sign-less". Thus our definition of ϖ being always positive carries the implicit understanding that a numerically positive frequency value requires a different physical interpretation for ϖ and ϕ^{+}.

All subsequent equations will be formulated in line with the above definitions on the concept of frequency. Forcing oneself to accordingly rethink the common text-book equations related to the FT is an instructive endeavour yielding added levels of insight.

Furthermore, the concept of a "frequency sweep" is introduced as follows: the symbols $\vec{\varphi}$, $\vec{\varphi}$ and $\vec{\varphi}$ emphasize the fact that we take into account all values for the variable φ in the continuous sets $\overleftrightarrow{\boldsymbol{\omega}}$, $\vec{\boldsymbol{\omega}}$ and $\overleftarrow{\boldsymbol{\omega}}$, respectively. Thus we may think of $\vec{\varphi}$, $\vec{\varphi}$ and $\vec{\varphi}$ as representing a "sweep" of the frequency value from $-\infty$ to $+\infty$, from 0 to $+\infty$, and from 0 to $-\infty$, respectively, in phasor-frequency space. Likewise, the symbol $\vec{\varpi}$ designates a trigonometric-frequency "sweep" of ϖ from 0 to $+\infty$ in the set $\vec{\boldsymbol{\varpi}}$. Frequency-sweep notation will be used, when needed for emphasis, in expressions describing temporal harmonic waves as well as in the function names of FT-spectra. For example, when stressing that we are considering the set of all sinusoid harmonic waves $\underline{h}_{\varpi}(t)$ covering the entire trigonometric frequency range $\vec{\boldsymbol{\varpi}}$ where each wave lasts from $-\infty$ to $+\infty$, we write $\underline{h}^{\rightarrow}_{\varpi}(t)\big|_{\overleftrightarrow{\boldsymbol{t}}}$ = $\underline{A}(\vec{\varpi}) \cdot \cos[\vec{\varpi} t + \varphi^{\circ}(\vec{\varpi})]\big|_{\overleftrightarrow{\boldsymbol{t}}}$. Similarly, the set of all perpetual harmonic phasors in the infinite phasor-frequency range $\overleftrightarrow{\boldsymbol{\omega}}$ can be written as $\underline{h}^{\leftrightarrow}_{\vec{\varphi}}(t)\big|_{\overleftrightarrow{\boldsymbol{t}}}$ = $\underline{A}(\vec{\varphi}) \cdot \exp(i \cdot \vec{\varphi} t)\big|_{\overleftrightarrow{\boldsymbol{t}}}$. If however we want to focus on a harmonic wave at a given frequency value we write $\underline{h}_{\varpi@}(t) = \underline{A}_{@} \cdot \cos(\varpi@ t + \varphi^{\circ}_{@})$ or $\underline{h}_{\varphi_{@}}(t) = \underline{A}_{@} \cdot \exp(i \cdot \varphi_{@} t)$.

2.3 FT-contextualized function notation in the time and frequency dimensions

The ways in which the forward and inverse FT operators $\underline{\overrightarrow{\mathrm{FT}}}$ and $\underline{\overleftarrow{\mathrm{FT}}}$ connect a t-D function $\underline{f}(t)$ and its ω-D counterpart $\underline{F}(\omega)$ are denoted as $\underline{F}(\omega) = \underline{\overrightarrow{\mathrm{FT}}}\{\underline{f}(t)\}$ and $\underline{f}(t) = \underline{\overleftarrow{\mathrm{FT}}}\{\underline{F}(\omega)\}$, respectively. The so-called Fourier Inversion Theorem (FIT) (see below) states that the FT is an invertible transform: there is a one-to-one correspondence between all temporal and spectral function pairs that we consider here; to reflect this concept the forward and backward FT operator symbols are combined into the symbol $\underline{\overleftrightarrow{\mathrm{FT}}}$, so the link between the two dimensions can be depicted as: $\underline{f}(t) \xleftrightarrow{\overleftrightarrow{\mathrm{FT}}} \underline{F}(\omega)$.

In the time dimension we generally consider complex-valued temporal functions $\underline{f}(t)\big|_{\overleftrightarrow{\boldsymbol{t}}}$ on a real timeline, with the operative duration $\overleftrightarrow{\boldsymbol{t}}$ corresponding to the time frame over which the (forward) FT is performed. This means that $\underline{f}(t)\big|_{\overleftrightarrow{\boldsymbol{t}}}$ = "*something*" for $t \in \overleftrightarrow{\boldsymbol{t}}$ (that "something" may contain values of $\underline{f}(t)$ = zero besides nonzero values) but $\underline{f}(t)\big|_{\overleftrightarrow{\boldsymbol{t}}}$ = "*nothing*" (i.e., $\underline{f}(t)$ is either zero or nonexistent) for all $t \notin \overleftrightarrow{\boldsymbol{t}}$. All $\underline{f}(t)\big|_{\overleftrightarrow{\boldsymbol{t}}}$ functions that are of practical relevance to us can be Fourier-transformed, i.e., their spectrum exists and $\underline{f}(t)\big|_{\overleftrightarrow{\boldsymbol{t}}}$ can be recovered exactly

(for all practical purposes) from its spectrum, and all nonperiodic temporal functions labeled here as $\underline{f}(t)\big|_{\boldsymbol{t}}$ have the extra feature that their function values are zero in the infinite past and in the infinite future - a property needed so that they can be absolute integrable (the integral of $\left|\underline{f}(t)\big|_{\boldsymbol{t}}\right|$ over all t exists and is finite) which in turn is required for the spectrum of $\underline{f}(t)\big|_{\boldsymbol{t}}$ to exist in a form that contains only finite function values.

In the frequency dimension, when thinking of the spectrum $\underline{F}(\omega)\big|_{\boldsymbol{\omega}}$ as a mathematical object that has been obtained by the forward FT of $\underline{f}(t)\big|_{\boldsymbol{t}}$, it will prove useful to extend the name $\underline{F}(\omega)\big|_{\boldsymbol{\omega}}$ such that it can remind us of what specific operative temporal interval $\underline{f}(t)\big|_{\boldsymbol{t}}$ existed on. To that end the temporal function's operative interval $\boldsymbol{t}$ will be indicated in the spectrum's name as $\underline{F}_{\boldsymbol{t}}(\omega)\big|_{\boldsymbol{\omega}}$. Although $\underline{f}(t)\big|_{\boldsymbol{t}}$ and $\underline{F}_{\boldsymbol{t}}(\omega)\big|_{\boldsymbol{\omega}}$ are equivalent representations of the same mathematical object, $\underline{f}(t)\big|_{\boldsymbol{t}}$, we may regard $\underline{f}(t)\big|_{\boldsymbol{t}}$ as the primary mathematical entity since the physical events that we are interested in and which are modelled by $\underline{f}(t)\big|_{\boldsymbol{t}}$ actually "happen" in time. A function $\underline{f}(t)\big|_{\boldsymbol{t}}$ describing an "event" of duration $\boldsymbol{t}$ is defined as zero outside of $\boldsymbol{t}$, and in that sense we can ignore all $t \notin \boldsymbol{t}$ without actually having to "sweep through" the natural domain from $t = -\infty$ to $t = +\infty$. However, most $\underline{F}_{\boldsymbol{t}}(\omega)\big|_{\boldsymbol{\omega}}$ spectra that we will encounter have an infinite operative interval $\overset{\Leftrightarrow}{\boldsymbol{\omega}}$ or $\overset{\Rightarrow}{\boldsymbol{\varpi}}$ such that the spectrum is continuous and has nonzero function values (allowing for zero values at points where $\underline{F}_{\boldsymbol{t}}(\omega)\big|_{\overset{\Leftrightarrow}{\boldsymbol{\omega}};\overset{\Rightarrow}{\boldsymbol{\varpi}}}$ may intercept or touch the ω axis) over its entire domain. Thus for the mathematical equivalence between $\underline{f}(t)\big|_{\boldsymbol{t}}$ and $\underline{F}_{\boldsymbol{t}}(\omega)\big|_{\overset{\Leftrightarrow}{\boldsymbol{\omega}};\overset{\Rightarrow}{\boldsymbol{\varpi}}}$ to hold, the forward FT must be performed by accounting for all possible frequencies in the infinite natural domains $\overset{\Leftrightarrow}{\boldsymbol{\omega}}$ or $\overset{\Rightarrow}{\boldsymbol{\varpi}}$, depending on whether we want to express the spectrum in ϕ-space or ϖ-space. To reflect this concept the notation for the pertinent frequency sweep (as described in the previous section) will be indicated in the spectrum's name. Thus, spectra with an infinite support are written as $\underline{F}_{\boldsymbol{t}}(\vec{\omega})\big|_{\overset{\Leftrightarrow}{\boldsymbol{\omega}}}$ or $\underline{F}_{\boldsymbol{t}}(\vec{\varpi})\big|_{\overset{\Rightarrow}{\boldsymbol{\varpi}}}$.

An ϕ-space spectrum can always be equated with a suitable, complex combination of two ϖ-space spectra, one obtained through the "cosine transform" and one through the "sine transform" of $\underline{f}(t)\big|_{\boldsymbol{t}}$. Phasor representation usually being more convenient to work with, herein preference will generally be given to ϕ-space formalism (however, the trigonometric form will also be discussed to some extent since this gives some interesting added insight into the nature of the FT). In all, the FT relationship will most generally be thought of as:

$$\underline{f}(t)\Big|_{t} \overset{\overset{\leftrightarrow}{\mathrm{FT}}}{\longleftrightarrow} \underline{F}_{t}(\vec{\omega})\Big|_{\omega} \tag{3}$$

Finally, we will find that some spectra of the type $\underline{F}_{t}(\vec{\omega})\Big|_{\omega}^{\Leftrightarrow}$ are centred about a specific frequency $\omega_{@}$. Although in such cases the value of $\omega_{@}$ is contained in the relevant rule, $\mathrm{Rule}\overline{\big\langle \omega - \omega_{@} \big|}_{\omega}^{\Leftrightarrow}$, it will often prove informative to indicate $\omega_{@}$ in the function's name as well, which will be done by writing $\underline{F}_{t}(\vec{\omega}_{\succ \omega_{@}})\Big|_{\omega}^{\Leftrightarrow} = \mathrm{Rule}\overline{\big\langle \omega - \omega_{@} \big|}_{\omega}^{\Leftrightarrow}$.

3. "Enhanced" Fourier-Transform equations

3.1 Euler's identities

With the above considerations in mind we can reformulate Euler's identities as follows. For a given frequency we have

$$\underline{A}_{@} \cdot e^{i \cdot \omega_{@} t}\Big|_{t} = {}^{\Re}A_{@} \cdot e^{i \cdot \omega_{@} t}\Big|_{t} + i \cdot {}^{\Im}A_{@} \cdot e^{i \cdot \omega_{@} t}\Big|_{t} = \underline{A}_{@} \cdot \cos(\varpi_{@} t)\Big|_{t}^{(\omega_{@}^{\pm})} \pm i \cdot \underline{A}_{@} \cdot \sin(\varpi_{@} t)\Big|_{t} \tag{4}$$

$$\underline{A}_{@} \cdot \cos(\varpi_{@} t)\Big|_{t} = 0.5\underline{A}_{@} \cdot e^{i \cdot \omega_{@}^{+} t}\Big|_{t} + 0.5\underline{A}_{@} \cdot e^{-i \cdot \omega_{@}^{+} t}\Big|_{t} \tag{5}$$

$$\underline{A}_{@} \cdot \sin(\varpi_{@} t)\Big|_{t} = -i \cdot 0.5\underline{A}_{@} e^{i \cdot \omega_{@}^{+} t}\Big|_{t} + i \cdot 0.5\underline{A}_{@} \cdot e^{-i \cdot \omega_{@}^{+} t}\Big|_{t} \tag{6}$$

On the other hand the entire positive, entire negative, or complete positive and negative phasor frequency range can be embraced via positive trigonometric frequency notation which may be represented by writing Equations (4) – (6) as

$$\underline{A}(\vec{\omega}) \cdot e^{i \cdot \overset{\leftrightarrow}{\omega} t}\Big|_{t} = {}^{\Re}A(\vec{\omega}) \cdot e^{i \cdot \overset{\leftrightarrow}{\omega} t}\Big|_{t} + i \cdot {}^{\Im}A(\vec{\omega}) \cdot e^{i \cdot \overset{\leftrightarrow}{\omega} t}\Big|_{t} = \underline{A}(\vec{\varpi}) \cdot \cos(\vec{\varpi} t)\Big|_{t}^{(\omega^{\pm})} \pm i \cdot \underline{A}(\vec{\varpi}) \cdot \sin(\vec{\varpi} t)\Big|_{t} \tag{7}$$

$$\underline{A}(\vec{\varpi}) \cdot \cos(\vec{\varpi} t)\Big|_{t} = 0.5\underline{A}(\vec{\omega}) \cdot e^{i \cdot \overset{\rightarrow}{\omega} t}\Big|_{t} + 0.5\underline{A}(\vec{\omega}) \cdot e^{i \cdot \overset{\leftarrow}{\omega} t}\Big|_{t} \tag{8}$$

$$\underline{A}(\vec{\varpi}) \cdot \sin(\vec{\varpi} t)\Big|_{t} = -i \cdot 0.5\underline{A}(\vec{\omega}) \cdot e^{i \cdot \overset{\rightarrow}{\omega} t}\Big|_{t} + i \cdot 0.5\underline{A}(\vec{\omega}) \cdot e^{i \cdot \overset{\leftarrow}{\omega} t}\Big|_{t} \tag{9}$$

These equations reflect the concept that we can describe a harmonic phasor's frequency and sign of rotation in both phasor notation and complex trigonometric form, so long as in the latter case we use both the (real) cosine and (imaginary) sine waves - both of which have algebraically positive ϖ frequencies. There is no contradiction in the fact that we can obtain a negative phasor frequency ω^{-} by combining a real and an imaginary sinusoid, both of

which have positive ϖ trigonometric frequencies. Equations (5,6) and (8,9) show that while a given sinusoid has no frequency sign ambiguity (it is always positive) in ϖ-space, it exhibits a sign duality when expressed in φ-space, i.e., a sinusoid is always equated with the sum of two phasors, one with a positive and one with a negative frequency.

3.2 Interrelationships in temporal and spectral representations

In our formalism a key stepping stone to the FT involves the consideration of Euler's identities from both a *t*-D and ω-D perspective. Let us take a harmonic phasor $\underline{A}_{@} \cdot \exp(i \cdot \varphi_{@}^{\pm} t)\Big|_{t}$, called herein the *principle phasor*, which can be viewed as a vector rotating in time as shown in temporal representation for a positive phasor $\underline{A}_{@} \cdot \exp(i \cdot \varphi_{@}^{+} t)\Big|_{t}$ in Figure 1(a). We can also find it informative to represent our phasor as an ω-D entity $\underline{A}_{@}\big|_{\dot{\boldsymbol{\omega}}_{@}^{\pm}}$ as shown for $\underline{A}_{@} \cdot \exp(i \cdot \varphi_{@}^{+} t)\Big|_{t}$ in Figure 2(A). The temporal and spectral representations emphasize different aspects of the phasor: the temporal form shows its progression in time, from which the numerical value of $\varphi_{@}^{\pm}$ is not directly accessible, while the spectral representation displays the $\varphi_{@}^{\pm}$ value in φ-space, but offers no direct sense of the phasor's temporal behavior or of its operative interval t. Considering the principle phasor $\underline{A}_{@} \cdot \exp(i \cdot \varphi_{@}^{\pm} t)\Big|_{t}$ in the light of (4)–(6), we can formulate the following scheme:

Temporal:

$$\underline{A}_{@} \cdot e^{i \cdot \varphi_{@}^{\pm} t}\Big|_{t} = {}^{\Re}A_{@} \cdot e^{i \cdot \varphi_{@}^{\pm} t}\Big|_{t} + i \cdot {}^{\Im}A_{@} \cdot e^{i \cdot \varphi_{@}^{\pm} t}\Big|_{t} \tag{10t}$$

$$\underline{A}_{@} \cdot e^{i \cdot \varphi_{@}^{\pm} t}\Big|_{t} = \underbrace{\underline{A}_{@} \cdot \cos(\varpi_{@} t)\Big|_{t}}_{\|} + \underbrace{\pm i \cdot \underline{A}_{@} \cdot \sin(\varpi_{@} t)\Big|_{t}}_{\|} \tag{11t}$$

$$\underline{A}_{@} \cdot e^{i \cdot \varphi_{@}^{\pm} t}\Big|_{t} = \overbrace{\left\{\begin{array}{c} 0.5\underline{A}_{@} \cdot e^{i \cdot \varphi_{@}^{+} t}\Big|_{t} \\ + \\ 0.5\underline{A}_{@} \cdot e^{i \cdot \varphi_{@}^{-} t}\Big|_{t} \end{array}\right\}}^{} + \overbrace{\left\{\begin{array}{c} \pm 0.5\underline{A}_{@} \cdot e^{i \cdot \varphi_{@}^{+} t}\Big|_{t} \\ + \\ \mp 0.5\underline{A}_{@} \cdot e^{i \cdot \varphi_{@}^{-} t}\Big|_{t} \end{array}\right\}}^{} \tag{12t}$$

Spectral:

$$\underline{A}_{@}\big|_{\dot{\boldsymbol{\omega}}_{@}^{\pm}} = {}^{\Re}A_{@}\big|_{\dot{\boldsymbol{\omega}}_{@}^{\pm}} + i \cdot {}^{\Im}A_{@}\big|_{\dot{\boldsymbol{\omega}}_{@}^{\pm}} \tag{10s}$$

$$\underline{A}_{@}\big|_{\dot{\boldsymbol{\omega}}_{@}^{\pm}} \leftrightarrow \underbrace{\underline{A}_{@}\big|_{\dot{\boldsymbol{\varpi}}_{@}}}_{\updownarrow} \quad \& \quad \underbrace{\pm i \cdot \underline{A}_{@}\big|_{\dot{\boldsymbol{\varpi}}_{@}}}_{\updownarrow} \tag{11s}$$

$$\underline{A}_{@}\Big|_{\dot{\omega}_{@}^{\pm}} = \overbrace{\begin{Bmatrix} 0.5\underline{A}_{@}\Big|_{\dot{\omega}_{@}^{+}} \\ + \\ 0.5\underline{A}_{@}\Big|_{\dot{\omega}_{@}^{-}} \end{Bmatrix}} + \overbrace{\begin{Bmatrix} \pm 0.5\underline{A}_{@}\Big|_{\dot{\omega}_{@}^{+}} \\ + \\ \mp 0.5\underline{A}_{@}\Big|_{\dot{\omega}_{@}^{-}} \end{Bmatrix}} \qquad (12s)$$

Equation (10t) tells us that we can decompose the principle phasor into a sum of two component phasors, $^{\Re}A_{@} \cdot \exp(i \cdot \phi_{@}^{\pm} t)\Big|_{t}$ and $i \cdot {}^{\Im}A_{@} \cdot \exp(i \cdot \phi_{@}^{\pm} t)\Big|_{t}$, having at $t = 0$ a real and a pure imaginary amplitude, respectively, as shown for a positive phasor in Figure 1(b,c). In spectral form Eq. (10t) becomes (10s), and $^{\Re}A_{@}\Big|_{\dot{\omega}_{@}^{\pm}}$ and $i \cdot {}^{\Im}A_{@}\Big|_{\dot{\omega}_{@}^{\pm}}$ [Figure 2(B,C)]. This tells us that if we take the real and imaginary parts of the principle phasor's complex spectrum $\underline{A}_{@}\Big|_{\dot{\omega}_{@}^{\pm}}$ [Figure 2(A)], then the real spectrum $^{\Re}A_{@}\Big|_{\dot{\omega}_{@}^{\pm}}$ [Figure 2(B)] corresponds to the real-amplitude component $^{\Re}A_{@} \cdot \exp(i \cdot \phi_{@}^{\pm} t)\Big|_{t}$ [Figure 1(b)], and the imaginary part $i \cdot {}^{\Im}A_{@}\Big|_{\dot{\omega}_{@}^{\pm}}$ [Figures 2(C)] corresponds to the pure imaginary-amplitude component $i \cdot {}^{\Im}A_{@} \cdot \exp(i \cdot \phi_{@}^{\pm} t)\Big|_{t}$ [Figure 1(c)] of the principle phasor.

Neither the real, nor the imaginary spectrum are normally represented in a 3D complex amplitude vs. frequency graph as in Figure 2(B,C), but as 2D real-amplitude vs. frequency, or imaginary-amplitude vs. frequency graphs [the analogous graphical schemes for a negative principle phasor can be seen in the original work (Szantay, 2007)]. Although in this representation the concepts of "real" and "imaginary" spectrum may seem almost self-evident, one should note that the ordinate value at a given frequency of the real and imaginary spectrum is often erroneously understood as meaning the amplitude of a cosine and sine trigonometric harmonic wave, respectively.

Consider the decomposition of $\underline{A}_{@} \cdot \exp(i \cdot \phi_{@}^{\pm} t)\Big|_{t}$ into a cosine and a sine wave according to Eq. (11t). Because their amplitudes are related as $\underline{A}_{@}$ and $i \cdot \underline{A}_{@}$, the planes of oscillation of the cosine and sine wave are perpendicular to each other in the complex plane as shown in Figure 1(d,e). In spectral form [Expression (11s)] the respective complex amplitudes are depicted in ϖ-space according to Figure 2(D,E). Note that when we translate (11t) into spectral form, (11s) ceases to be an equality. This is because: a) the spectral amplitudes $\underline{A}_{@}\Big|_{\dot{\varpi}_{@}}$ for the cosine and sine components [Figure 2(D,E)] belong to two different mathematical "species", therefore they cannot simply be added; b) the spectrum of the principle phasor [Figure 2(A)] is a function in ϕ-space, while the spectra of its component sinusoids [Figures 2(D,E)] are functions in ϖ-space. (Again, this is a fundamental difference which can easily escape attention in the absence of a symbolism that differentiates between phasor and trigonometric frequencies.) Note also that we have no analogous problem with the relevant temporal forms of Figures 1(a,d,e), since they have a common argument: time. Thus the amplitudes depicted in Figure 2(D,E) do not add up to those shown in Figure 2(A).

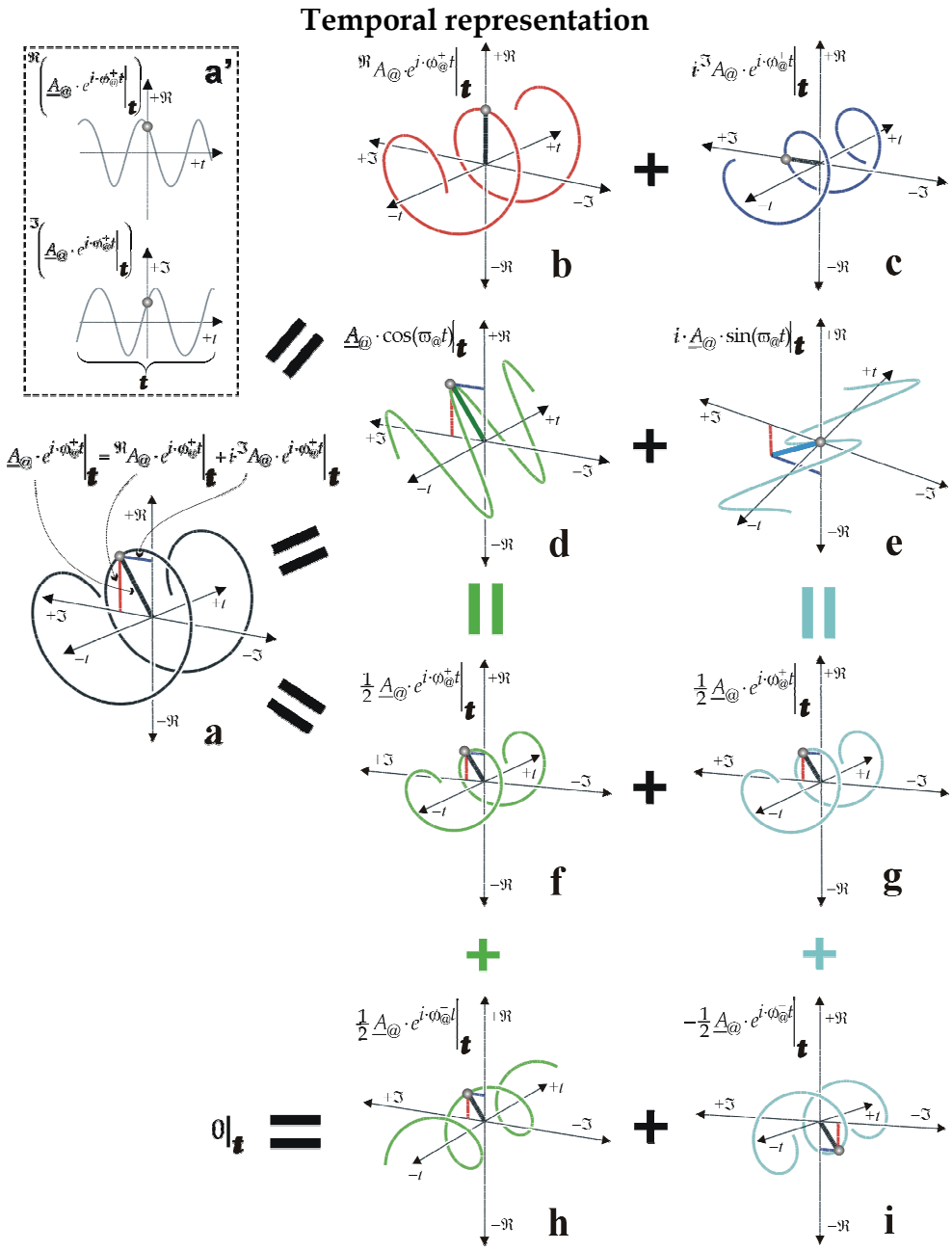


Fig. 1. Graphical representation of Equations (10t)-(12t) for a positive principle phasor. For added clarity the side-view projections of the principle phasor onto the real and imaginary axis are shown in (a')

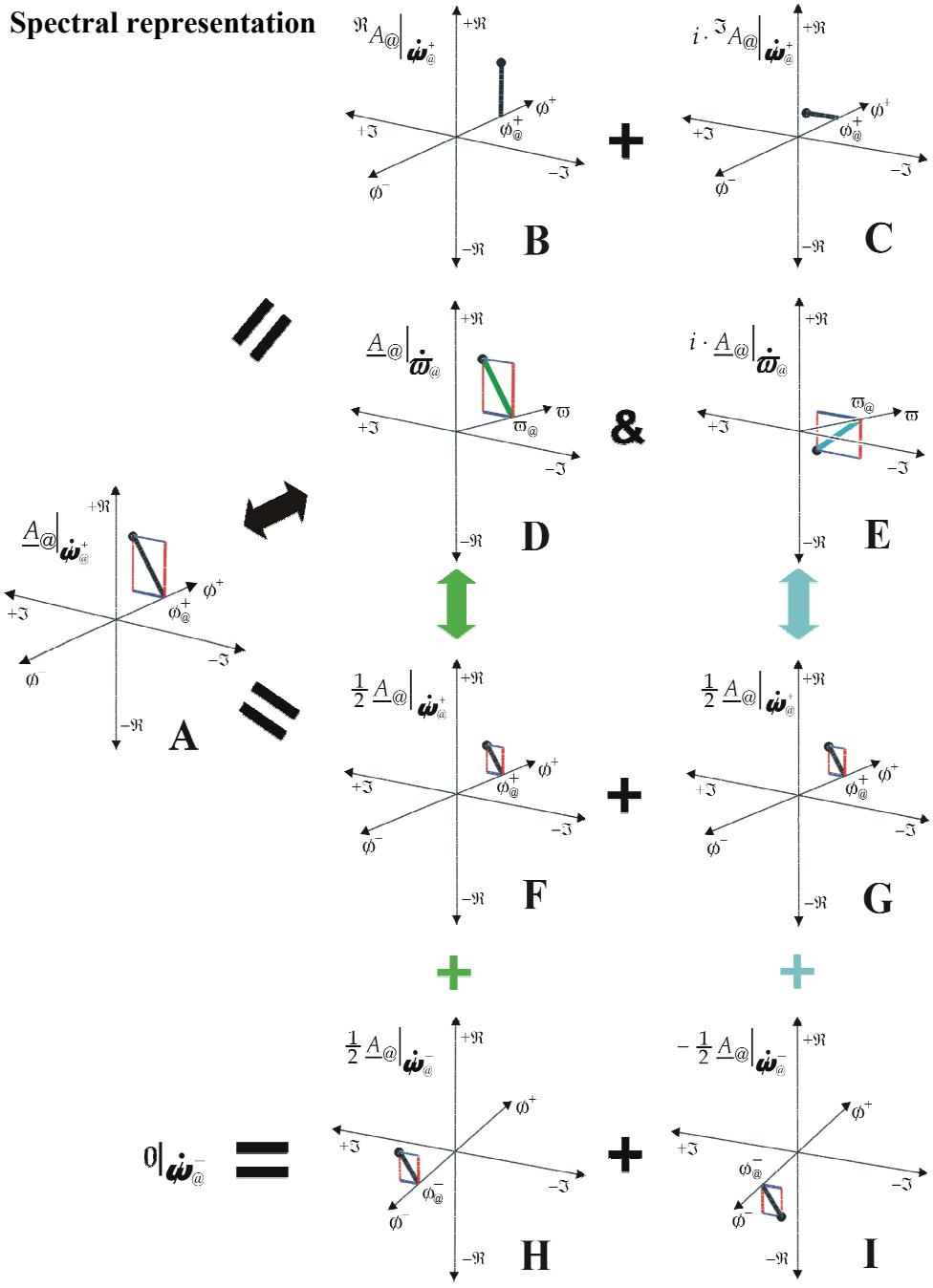


Fig. 2. The spectral representational analogue of Figure 1 according to Equations (10S)-(12S)

In order to bring $\underline{A}_{@}\big|_{\dot{\omega}_{@}^{\pm}}$, $\underline{A}_{@}\big|_{\dot{\varpi}_{@}}$, and $i\cdot\underline{A}_{@}\big|_{\dot{\varpi}_{@}}$ onto a "common ground" we need to translate the ϖ-space spectra into φ-space. In the temporal domain this means that the principle phasor's cosine component [Figure 1(d)] and sine component [Figure 1(e)] are decomposed into a sum of oppositely rotating phasors [Figure 1(f,h)] according to (12t). The decomposition of the cosine term reflects Equations (5) while that of the sine term corresponds to (6) multiplied by "i". If we now add up the four phasors in (12t) we re-gain our principle phasor $\underline{A}_{@}\cdot\exp(i\cdot\varphi_{@}^{\pm}t)\big|_{t}$ [Figures 1(a)]: the two negative phasor-frequency components have identical amplitudes but are in opposite phase [Figures 1(h,i)] and hence cancel each other at every point in time, while the two positive phasor-frequency components, both having an amplitude $0.5\underline{A}_{@}$, are in-phase [Figure 1(f,g)] and thus reinforce each other to give the original principle phasor.

In spectral notation the relationships of (12t) are expressed according to (12s) and the pertinent φ-space spectra are shown in Figure 2(F,G,H,I). In analogy to the correlation between the left and right sides of (11s), there is a correlation, but not an equation, between the pertinent terms on the right side of (11s) and (12s).

By adding up the four amplitudes in [12s] we obtain the principle phasor's spectrum $\underline{A}_{@}\big|_{\dot{\omega}_{@}^{\pm}}$ [Figures 2(A)]. We may think of these "secondary-phasor" spectra [Figures 2(F,G,H,I)] as "interfaces" that allow us to connect the principle phasor's φ-space spectrum [Figures 2(A)] with its ϖ-space spectra [Figures 2(D,E)].

From a broader viewpoint, the whole basis of the FT rests on the idea that a temporal function can be constructed from suitable harmonic "principle phasors". Any given point on the complex FT-spectrum will represent a constituent harmonic principle phasor. It is according to the above scheme that we can conceptually envisage the way the spectrum of a temporal function (or signal) is obtained.

3.3 The Fourier Transform

The FT is based on the theorem that $\underline{f}(t)\big|_{t}$ can be uniquely expressed in terms of its projections onto an infinite frequency set of basis phasors $\underline{h}_{\vec{\varphi}}^{\leftrightarrow}(t)\big|_{\overset{\leftrightarrow}{t}} = \underline{A}(\vec{\varphi})\cdot\exp(i\cdot\vec{\varphi}t)\big|_{\overset{\leftrightarrow}{t}}$. In general, the infinite set of unit phasors $\exp(i\cdot\vec{\varphi}t)\big|_{\overset{\leftrightarrow}{t}}$ constitutes a unitary orthogonal basis set for all Fourier-transformable $\underline{f}(t)\big|_{t}$ functions, which can therefore be represented as a weighted sum of the unit harmonic waves. The spectrum $\underline{F}_{t}(\vec{\varphi})\big|_{\overset{\leftrightarrow}{\omega}}$ specifies the weighting coefficient for the unit basis phasors at all $\varphi\in\overset{\leftrightarrow}{\omega}$ frequencies. Using both trigonometric and phasor notation [cf. Equations (7)-(9)], the essence of the FT can conceptually be written as:

$$\underline{f(t)}\Big|_{\overleftrightarrow{t}} = \int_{0}^{\infty} \underline{h}^{\rightarrow}_{\varpi}(t)\Big|_{\overleftrightarrow{t}} \cdot d\varpi = \int_{-\infty}^{\infty} \underline{h}^{\leftrightarrow}_{\phi}(t)\Big|_{\overleftrightarrow{t}} \cdot d\phi \tag{13}$$

Employing the nomenclature and concepts outlined above, the Fourier integrals can be expressed as follows:

$$\underline{f(t)}\Big|_{\overleftrightarrow{t}} = \int_{0}^{\infty} \underline{h}^{\rightarrow}_{\varpi}(t)\Big|_{\overleftrightarrow{t}} \cdot d\varpi = \underbrace{\frac{1}{\pi} \cdot \int_{0}^{\infty} \underline{F}^{\cos}_{\overleftrightarrow{t}}(\vec{\varpi})\Big|_{\overrightarrow{\varpi}} \cdot \cos(\vec{\varpi} t)\Big|_{\overleftrightarrow{t}} \cdot d\varpi}_{\overleftarrow{\mathrm{FT}}(\cos)\left\langle \underline{F}^{\cos}_{\overleftrightarrow{t}}(\vec{\varpi})\Big|_{\overrightarrow{\varpi}} = \underline{f(t)}^{\cos}\right|_{\overleftrightarrow{t}}} + \underbrace{\frac{1}{\pi} \cdot \int_{0}^{\infty} \underline{F}^{\sin}_{\overleftrightarrow{t}}(\vec{\varpi})\Big|_{\overrightarrow{\varpi}} \cdot \sin(\vec{\varpi} t)\Big|_{\overleftrightarrow{t}} \cdot d\varpi}_{\overleftarrow{\mathrm{FT}}(\sin)\left\langle \underline{F}^{\sin}_{\overleftrightarrow{t}}(\vec{\varpi})\Big|_{\overrightarrow{\varpi}} = \underline{f(t)}^{\sin}\right|_{\overleftrightarrow{t}}} = \tag{14a}$$

$$= \int_{-\infty}^{\infty} \underline{h}^{\leftrightarrow}_{\phi}(t)\Big|_{\overleftrightarrow{t}} \cdot d\phi = \underbrace{\frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} \underline{F}_{\overleftrightarrow{t}}(\vec{\phi})\Big|_{\overrightarrow{\omega}} \cdot e^{i \cdot \overleftrightarrow{\phi} t}\Big|_{\overleftrightarrow{t}} \cdot d\phi}_{\overleftarrow{\underline{\mathrm{FT}}}\left\langle \underline{F}_{\overleftrightarrow{t}}(\vec{\phi})\Big|_{\overrightarrow{\omega}} = \underline{f(t)}\right|_{\overleftrightarrow{t}}} \tag{14b}$$

$$\underline{F}^{\cos}_{\overleftrightarrow{t}}(\vec{\varpi})\Big|_{\overrightarrow{\varpi}} = \int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\overleftrightarrow{t}} \cdot \cos(\vec{\varpi} t)\Big|_{\overleftrightarrow{t}} \cdot dt = \int_{\overleftrightarrow{t}} \underline{f(t)}\Big|_{\overleftrightarrow{t}} \cdot \cos(\vec{\varpi} t)\Big|_{\overleftrightarrow{t}} \cdot dt = \overrightarrow{\mathrm{FT}}(\cos)\left\langle \underline{f(t)}\Big|_{\overleftrightarrow{t}}\right. \tag{15a$^{\cos}$}$$

$$\underline{F}^{\sin}_{\overleftrightarrow{t}}(\vec{\varpi})\Big|_{\overrightarrow{\varpi}} = \int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\overleftrightarrow{t}} \cdot \sin(\vec{\varpi} t)\Big|_{\overleftrightarrow{t}} \cdot dt = \int_{\overleftrightarrow{t}} \underline{f(t)}\Big|_{\overleftrightarrow{t}} \cdot \sin(\vec{\varpi} t)\Big|_{\overleftrightarrow{t}} \cdot dt = \overrightarrow{\mathrm{FT}}(\sin)\left\langle \underline{f(t)}\Big|_{\overleftrightarrow{t}}\right. \tag{15a$^{\sin}$}$$

$$\underline{F}_{\overleftrightarrow{t}}(\vec{\phi})\Big|_{\overrightarrow{\omega}} = \int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\overleftrightarrow{t}} \cdot e^{-i \cdot \overleftrightarrow{\phi} t}\Big|_{\overleftrightarrow{t}} \cdot dt = \int_{\overleftrightarrow{t}} \underline{f(t)}\Big|_{\overleftrightarrow{t}} \cdot e^{-i \cdot \overleftrightarrow{\phi} t}\Big|_{\overleftrightarrow{t}} \cdot dt = \overrightarrow{\underline{\mathrm{FT}}}\left\langle \underline{f(t)}\Big|_{\overleftrightarrow{t}}\right. \tag{15b}$$

Equations (15) and (14) express the forward and inverse FT, respectively. In the spectra $\underline{F}^{\cos}_{\overleftrightarrow{t}}(\vec{\varpi})\Big|_{\overrightarrow{\varpi}}$, $\underline{F}^{\sin}_{\overleftrightarrow{t}}(\vec{\varpi})\Big|_{\overrightarrow{\varpi}}$ and $\underline{F}_{\overleftrightarrow{t}}(\vec{\phi})\Big|_{\overrightarrow{\omega}}$ each ϕ or ϖ value corresponds to a pure harmonic wave and the unit basis functions $\exp(i \cdot \vec{\phi} t)\Big|_{\overleftrightarrow{t}}$ are declared on an infinite time scale $\overleftrightarrow{t}$, which means that irrespective of what operative temporal interval $\underline{f(t)}\Big|_{\overleftrightarrow{t}}$ "exists" upon, it can be decomposed into an infinite set of eternal harmonic waves. The eternity of the unitary orthogonal basis set $\exp(i \cdot \vec{\phi} t)\Big|_{\overleftrightarrow{t}}$ is a generally not fully appreciated, but fundamentally important aspect of the FT, easily escaping attention in the standard mathematical symbolism. For example, in the conventional notation Equation (14b) is simply written as $f(t) = 1/2\pi \cdot \int_{-\infty}^{\infty} F(\omega) \cdot \exp(i \cdot \omega t) \cdot d\omega$, and this expression offers little in the way of showing us that even if $f(t)$ is finite in time, the basis phasors $e^{i \cdot \omega t}$ are perpetual. Further insight may be gained into the FT by noting that the right-hand side of Equation (14b) represents an infinite collection of eternal "principle phasors" on the phasor frequency scale. Thus, in analogy to Equations (10)-(12) we can formulate the following scheme:

Temporal:

$$\underbrace{\frac{1}{2\pi}\cdot\int_{-\infty}^{\infty}\underline{F}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\cdot e^{i\cdot\overleftrightarrow{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\omega}_{\underline{f}(t)\big|_{\boldsymbol{t}}} =$$

$$= \underbrace{\Re\left(\frac{1}{2\pi}\cdot\int_{-\infty}^{\infty}\underline{F}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\cdot e^{i\cdot\overleftrightarrow{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\omega\right)}_{\Re f(t)\big|_{\boldsymbol{t}}} + \underbrace{i\cdot\Im\left(\frac{1}{2\pi}\cdot\int_{-\infty}^{\infty}\underline{F}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\cdot e^{i\cdot\overleftrightarrow{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\omega\right)}_{i\cdot\Im f(t)\big|_{\boldsymbol{t}}} \tag{16t}$$

$$= \underbrace{\frac{1}{\pi}\cdot\int_{0}^{\infty}\underline{F}^{\cos}_{\boldsymbol{t}}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}}\cdot\cos(\vec{\varpi}t)\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\varpi}_{\underline{f}(t)^{\cos}\big|_{\boldsymbol{t}}} + \underbrace{\frac{1}{\pi}\cdot\int_{0}^{\infty}\underline{F}^{\sin}_{\boldsymbol{t}}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}}\cdot\sin(\vec{\varpi}t)\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\varpi}_{\underline{f}(t)^{\sin}\big|_{\boldsymbol{t}}} \tag{17t}$$

$$\Big\|\qquad\qquad\Big\|$$

$$= \overbrace{\left\{\begin{array}{c}\frac{1}{2\pi}\cdot\int_{0}^{\infty}\underline{F}^{\cos}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\cdot e^{i\cdot\vec{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\omega \\ + \\ \frac{1}{2\pi}\cdot\int_{-\infty}^{0}\underline{F}^{\cos}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\cdot e^{i\cdot\overleftarrow{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\omega\end{array}\right\}}^{} + \overbrace{\left\{\begin{array}{c}\frac{1}{2\pi}\cdot\int_{0}^{\infty}\underline{F}^{\sin}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\cdot e^{i\cdot\vec{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\omega \\ + \\ \frac{1}{2\pi}\cdot\int_{-\infty}^{0}\underline{F}^{\sin}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\cdot e^{i\cdot\overleftarrow{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot d\omega\end{array}\right\}}^{} \tag{18t}$$

Spectral:

$$\underline{F}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = {}^{\Re}F_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} + i\cdot{}^{\Im}F_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} \tag{16s}$$

$$\underline{F}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} \leftrightarrow \underbrace{\underline{F}^{\cos}_{\boldsymbol{t}}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}}}_{\updownarrow} \;\&\; \underbrace{\underline{F}^{\sin}_{\boldsymbol{t}}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}}}_{\updownarrow} \tag{17s}$$

$$\underline{F}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = \overbrace{\left\{\begin{array}{c}\underline{F}^{\cos}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} \\ + \\ \underline{F}^{\cos}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\end{array}\right\}}^{} + \overbrace{\left\{\begin{array}{c}\underline{F}^{\sin}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} \\ + \\ \underline{F}^{\sin}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}\end{array}\right\}}^{} \tag{18s}$$

From Equations (4) and (5) we can express the forward FT for the phasor-space spectral form also as:

$$\overbrace{\int_{-\infty}^{\infty}\underline{f}(t)\Big|_{\boldsymbol{t}}\cdot e^{-i\cdot\overleftrightarrow{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot dt}^{\overrightarrow{\mathrm{FT}}} = \underbrace{\Re\left(\int_{-\infty}^{\infty}\underline{f}(t)\Big|_{\boldsymbol{t}}\cdot e^{-i\cdot\overleftrightarrow{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot dt\right)}_{\Re F_{\boldsymbol{t}}(\vec{\omega})\big|_{\boldsymbol{\omega}}} + \underbrace{i\cdot\Im\left(\int_{-\infty}^{\infty}\underline{f}(t)\Big|_{\boldsymbol{t}}\cdot e^{-i\cdot\overleftrightarrow{\omega}\,t}\Big|_{\Leftrightarrow_{\boldsymbol{t}}}\cdot dt\right)}_{i\cdot\Im F_{\boldsymbol{t}}(\vec{\omega})\big|_{\boldsymbol{\omega}}} \tag{19s}$$

$$\underline{F}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}$$

$$= \overbrace{\underbrace{\int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\mathbf{t}} \cdot \cos(\vec{\varpi}t)\Big|_{\mathbf{t}}^{\Leftrightarrow} \cdot dt}_{\underline{F}_{\mathbf{t}}^{\cos}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}}}}^{\overrightarrow{\mathrm{FT}}_{(\cos)}} \overset{(\omega^{\pm})}{\mp} \overbrace{i \cdot \underbrace{\int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\mathbf{t}} \cdot \sin(\vec{\varpi}t)\Big|_{\mathbf{t}}^{\Leftrightarrow} \cdot dt}_{\underline{F}_{\mathbf{t}}^{\sin}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}}}}^{\overrightarrow{\mathrm{FT}}_{(\sin)}} \tag{20s}$$

$$= \left\{ \begin{array}{l} \overbrace{\underbrace{0.5 \cdot \int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\mathbf{t}} \cdot e^{i \cdot \vec{\omega} t}\Big|_{\mathbf{t}}^{\Leftrightarrow} \cdot dt}_{0.5 \cdot \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}}}^{} \\ + \\ \underbrace{0.5 \cdot \int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\mathbf{t}} \cdot e^{i \cdot \overleftarrow{\omega} t}\Big|_{\mathbf{t}}^{\Leftrightarrow} \cdot dt}_{0.5 \cdot \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}} \end{array} \right\} \overset{(\omega^{\pm})}{\mp} i \cdot \left\{ \begin{array}{l} \overbrace{\underbrace{-i \cdot 0.5 \cdot \int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\mathbf{t}} \cdot e^{i \cdot \vec{\omega} t}\Big|_{\mathbf{t}}^{\Leftrightarrow} \cdot dt}_{-i \cdot 0.5 \cdot \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}}}^{} \\ + \\ \underbrace{i \cdot 0.5 \cdot \int_{-\infty}^{\infty} \underline{f(t)}\Big|_{\mathbf{t}} \cdot e^{i \cdot \overleftarrow{\omega} t}\Big|_{\mathbf{t}}^{\Leftrightarrow} \cdot dt}_{i \cdot 0.5 \cdot \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}} \end{array} \right\} \tag{21s}$$

In Equations (16) - (21) the relationships between the various ϖ-space and ω-space spectral forms are as follows:

$$\left. \begin{array}{l} \underline{F}_{\mathbf{t}}^{\cos}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = \underline{F}_{\mathbf{t}}^{\cos}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = \underline{F}_{\mathbf{t}}^{\cos}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}}; \underline{F}_{\mathbf{t}}^{\sin}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = -i \cdot \underline{F}_{\mathbf{t}}^{\sin}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}}; \underline{F}_{\mathbf{t}}^{\sin}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = i \cdot \underline{F}_{\mathbf{t}}^{\sin}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}} \\ \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = \underline{F}_{\mathbf{t}}^{\cos}(\vec{\omega})\Big|_{\boldsymbol{\omega}} + \underline{F}_{\mathbf{t}}^{\sin}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = \underline{F}_{\mathbf{t}}^{\cos}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}} - i \cdot \underline{F}_{\mathbf{t}}^{\sin}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}} \\ \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = \underline{F}_{\mathbf{t}}^{\cos}(\vec{\omega})\Big|_{\boldsymbol{\omega}} + \underline{F}_{\mathbf{t}}^{\sin}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = \underline{F}_{\mathbf{t}}^{\cos}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}} + i \cdot \underline{F}_{\mathbf{t}}^{\sin}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}} \\ \underline{F}_{\mathbf{t}}^{\cos}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}} = 0.5 \cdot \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} + 0.5 \cdot \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = 0.5 \cdot \underline{F}_{\mathbf{t}}^{\cos}(\vec{\omega})\Big|_{\boldsymbol{\omega}} + 0.5 \cdot \underline{F}_{\mathbf{t}}^{\cos}(\vec{\omega})\Big|_{\boldsymbol{\omega}} \\ \underline{F}_{\mathbf{t}}^{\sin}(\vec{\varpi})\Big|_{\boldsymbol{\varpi}} = i \cdot 0.5 \cdot \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} - i \cdot 0.5 \cdot \underline{F}_{\mathbf{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}} = i \cdot 0.5 \cdot \underline{F}_{\mathbf{t}}^{\sin}(\vec{\omega})\Big|_{\boldsymbol{\omega}} - i \cdot 0.5 \cdot \underline{F}_{\mathbf{t}}^{\sin}(\vec{\omega})\Big|_{\boldsymbol{\omega}} \end{array} \right] \tag{22}$$

From this formalism one can draw a number of subtle insights into the FT as was detailed in the original papers which also included a discussion on a geometrical interpretation of the FT and on its capability to unravel the oscillating frequency components that are otherwise obscured in a complicated time signal (Szántay, 2007, 2008a, 2008b, 2008c). These aspects will not be elaborated here. Suffice to note only that if a signal, represented by the function $\underline{f(t)}\Big|_{\mathbf{t}}$, comprises of harmonic phasor components (such is the case with the NMR "free induction decay" except that it contains exponentially damped harmonic phasors), then the signal must be detected from two orthogonal directions simultaneously to be able to tell the sense of rotation of the pertinent phasors. Letting these directions correspond to the $+\Re$ and $+\Im$ axes we obtain the functions ${}^{\Re}f(t)\Big|_{\mathbf{t}}$ and $i \cdot {}^{\Im}f(t)\Big|_{\mathbf{t}}$ [cf. Equation (20t)] as the "side projections" of $\underline{f(t)}\Big|_{\mathbf{t}}$ which are analogous to the principle phasor's real and imaginary

"side-view" components as shown in Figure 1 (a'). Accordingly, in general $^{\Re}f(t)\big|_{t}$ and $i\cdot{}^{\Im}f(t)\big|_{t}$ both comprise of a mixture of cosine and sine waves, thus when we speak of the "complex forward" FT of $\underline{f}(t)\big|_{t}$, then this is actually accomplished through the FT of $^{\Re}f(t)\big|_{t}$ combined with the FT of $i\cdot{}^{\Im}f(t)\big|_{t}$. The relationship between the right-hand sides of (20s) and (21s), or (17s) and (18s), reflects the way in which the ϖ-space outputs of the cosine and sine transforms $\vec{\mathrm{FT}}_{(\cos)}$ and $\vec{\mathrm{FT}}_{(\sin)}$ assume a sign duality in φ-space. The addition of the "secondary" φ-space spectra $\underline{F}^{\cos}_{t}(\vec{\varphi})\big|_{\omega}$, $\underline{F}^{\cos}_{t}(\vec{\varphi})\big|_{\omega}$, $\underline{F}^{\sin}_{t}(\vec{\varphi})\big|_{\omega}$ and $\underline{F}^{\sin}_{t}(\vec{\varphi})\big|_{\omega}$ gives the final FT-spectrum $\underline{F}_{t}(\vec{\varphi})\big|_{\omega}$, i.e., only the $\vec{\mathrm{FT}}_{(\cos)}$ and $\vec{\mathrm{FT}}_{(\sin)}$ transforms together can properly identify the sign of the phasor frequency components within $\underline{f}(t)\big|_{t}$.

A few basic FT relationships, represented graphically as well as in terms of function rules written in the style discussed above are shown in Figure 3 (all of these formulas are of course available in conventional notation from standard textbooks).

4. Uncertainty principle(s)

4.1 The Heisenberg and Fourier Uncertainty Principles

With reference to the issue of whether or not the frequency of a time-limited sinusoid wave becomes inherently "fuzzy" due to the Uncertainty Principle (as was discussed in the Introduction), let us now take a closer look at the meaning of the term "Uncertainty Principle". One mental trap here is that for most people the phrase "Uncertainty Principle" invokes a single famous theorem, while there are actually several uncertainty principles that have come to constitute their independent branches of mathematics, physics and signal analysis theories. In the context of our present discussion we can speak of two main types of uncertainty principle: Heisenberg's Uncertainty Principle (HUP) which is a quantum-mechanical concept grounded in the physical world, and the Fourier Uncertainty Principle (FUP) which is a classical concept and stems from a mathematical necessity inherent to the way the FT works.

Within our framework of discussion the HUP can be written for the noncommuting variables time and frequency through the expression:

$$\Delta t \cdot \Delta_{\mathrm{sd}}\omega \geq \text{constant} \tag{23}$$

where the symbol Δ_{sd} denotes the standard deviation of frequency.

In the classical realm of time-frequency signal analysis there exists another fundamental inequality commonly referred to as the Fourier Uncertainty Principle (FUP). In qualitative terms the FUP states that the "broader" $\underline{\hat{f}}(t)\big|_{t}$ is in time, the "narrower" $\underline{F}_{t}(\vec{\varphi})\big|_{\omega}^{\Leftrightarrow}$ is in frequency and vice versa. Since there is no universally applicable mathematical or semantic

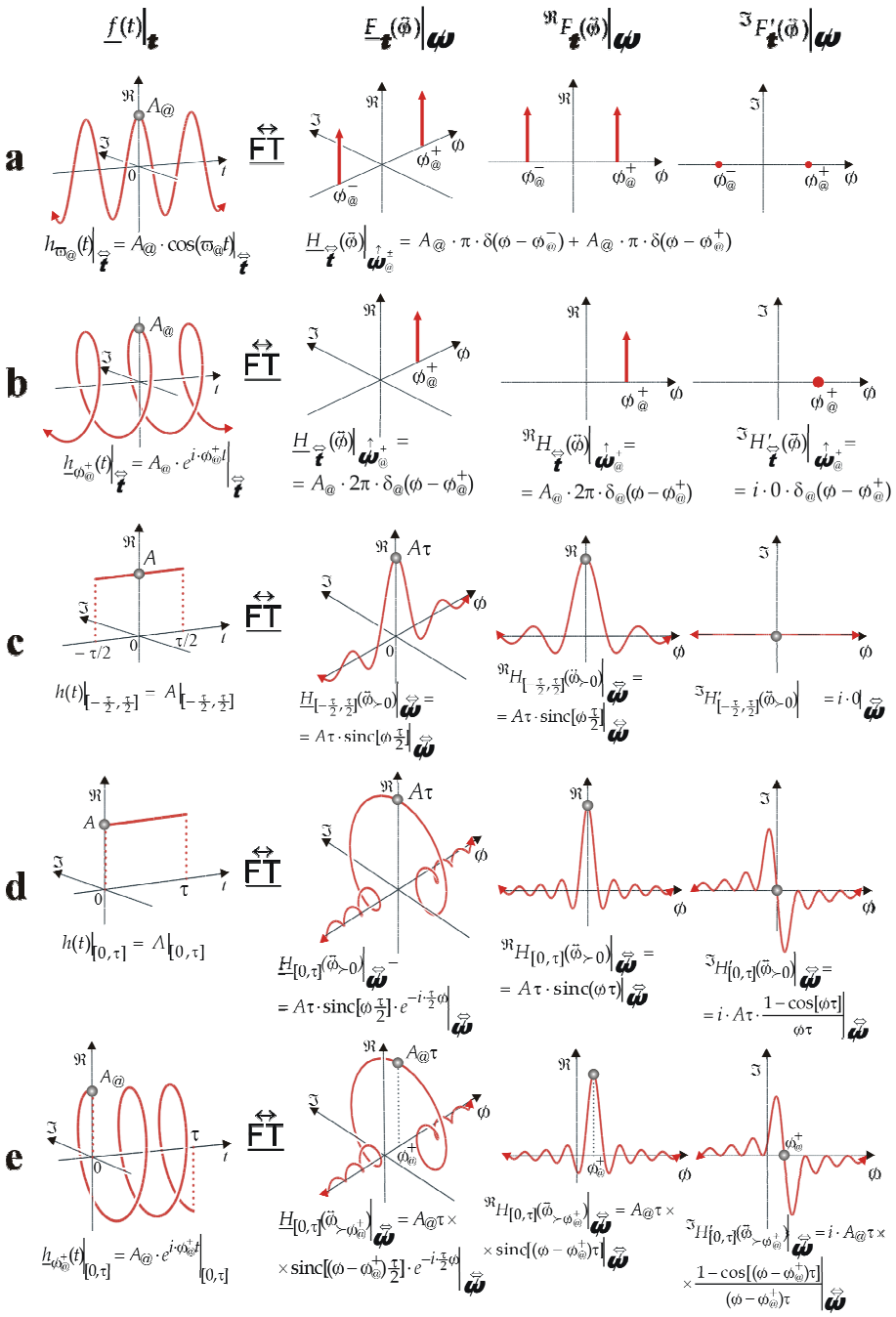


Fig. 3. Some special FT relationships

definition that would adequately quantify or delineate the concept of a function's "breadth" (Cohen, 1995 and Claerbout, 2004), we adopt from Claerbout the generic symbol "∧", herein called "*characteristic span*", connoting the "compressedness" or "elongatedness" of a function along its domain, the idea being that "∧" symbolizes any mathematical definition, including Δ_{sd}, that may be suitably applied to the specific function at hand. Using the symbol "¬" to denote "of", so "$\wedge x \neg f(x)\big|_{\boldsymbol{x}}$" means the characteristic span of $f(x)\big|_{\boldsymbol{x}}$, the FUP can thus be expressed as

$$\wedge t \neg \underline{\widehat{f}}(t)\Big|_{\boldsymbol{t}} \cdot \wedge \phi \neg \underline{F}_{\boldsymbol{t}}(\vec{\omega})\Big|_{\boldsymbol{\omega}}^{\Leftrightarrow} = \text{TBP} \geq \Xi \tag{24a}$$

$$\wedge t \cdot \wedge \phi = \text{TBP} \geq \Xi \tag{24b}$$

where TBP is the so-called "time-bandwidth product", Ξ is the minimum value of the TBP, and (24b) is just a short-hand for (24a). The actual values of TBP and Ξ are specific to the exact way in which $\wedge t \neg \underline{\widehat{f}}(t)\big|_{\boldsymbol{t}}$ and $\wedge \phi \neg \underline{F}_{\boldsymbol{t}}(\vec{\omega})\big|_{\boldsymbol{\omega}}^{\Leftrightarrow}$ have been defined for the particular $\underline{\widehat{f}}(t)\big|_{\boldsymbol{t}}$ and $\underline{F}_{\boldsymbol{t}}(\vec{\omega})\big|_{\boldsymbol{\omega}}^{\Leftrightarrow}$ functions at hand. The FUP is an equality in the sense that for any given absolute integrable function $\underline{\widehat{f}}(t)\big|_{\boldsymbol{t}}$ the TBP is constant, therefore increasing $\wedge t \neg \underline{\widehat{f}}(t)\big|_{\boldsymbol{t}}$ entails a concurrent decrease in $\wedge \phi \neg \underline{F}_{\boldsymbol{t}}(\vec{\omega})\big|_{\boldsymbol{\omega}}^{\Leftrightarrow}$ and vice versa; the FUP however is an inequality in the sense that the TBP must be greater than the value of Ξ. The FUP asserts nothing more than: a) a *t*-D function can only be "squeezed" at the expense of expanding its ω-D counterpart and vice versa; b) $\underline{\widehat{f}}(t)\big|_{\boldsymbol{t}}$ and $\underline{F}_{\boldsymbol{t}}(\vec{\omega})\big|_{\boldsymbol{\omega}}^{\Leftrightarrow}$ cannot both be made arbitrarily "narrow".

The FUP and HUP share a common mathematical form and terminology which is why, as it was pointed out by Cohen (Cohen, 1995), they are easily mixed up, especially since both are often written simply in the form "$\Delta t \cdot \Delta\omega \geq \text{constant}$". However, the physical interpretations of (23) and (24) are different: in the HUP the word "uncertainty" reflects probabilistic uncertainties in the measurement of quantum-mechanical physical observables, while the FUP is a deterministic theorem resulting from the innate nature of the FT, as was illustrated geometrically earlier (Szántay, 2008c).

The notion that by time-limiting a monochromatic sine wave we impose some "uncertainty" on its frequency stems either from confusing the HUP and the FUP, or from misunderstanding the FUP itself. With regard to the deterministic sine wave (such as the NMR pulse), it is obviously the FUP that is of relevance and not the HUP. As for the FUP, one reason it is often misinterpreted is because "uncertainty" is a dubious word with several possible interpretations in various situations, as was discussed in detail before (Szántay, 2008c). The concept of the characteristic span $\wedge x \neg \underline{\widehat{f}}(x)\big|_{\boldsymbol{x}}$ simply reflects "broadness" which is not a probabilistic quantity per se; just because a function has a large characteristic span, it is not "uncertain". In that sense, referring to (24) as an "Uncertainty Principle" is a misnomer, as was also emphasized by Cohen (Cohen, 1995).

4.2 The law of "Conjugate Physical Equivalence"

The Fourier Inversion Theorem (FIT) symbolized by (3) tells us that $\underline{f(t)}\big|_{\mathbb{t}}$ and $\underline{F_{\mathbb{t}}(\vec{\omega})}\big|_{\boldsymbol{\omega}}$, as being functions of the conjugate variables time and frequency, represent the same mathematical object $\underline{f(t)}\big|_{\mathbb{t}}$ expressed in two different "languages", temporal and spectral. The mathematical truth of the FIT implies that if $\underline{f(t)}\big|_{\mathbb{t}}$ is a model of some physical process, then whatever physical meaning we assign to $\underline{f(t)}\big|_{\mathbb{t}}$, we must assign the same physical meaning and information content to $\underline{F_{\mathbb{t}}(\vec{\omega})}\big|_{\boldsymbol{\omega}}$. For the sake of emphasis I will herein refer to this concept as the "Law of Conjugate Physical Equivalence" (LCPE).

Although, when phrased in this way the LCPE may seem rather self-evident, it is actually all too often overlooked when it comes to the physical interpretation of the mathematics of the FIT. The reason for this stems from the innate "linguistic" difference between $\underline{f(t)}\big|_{\mathbb{t}}$ and $\underline{F_{\mathbb{t}}(\vec{\omega})}\big|_{\boldsymbol{\omega}}$. The temporal form is intuitively more familiar since it reflects, and is more congruent with, our natural perception of physical events in the way they actually "happen" in time. The spectral "language" however is an artificial construct which is non-intuitive in the sense that when $\underline{f(t)}\big|_{\mathbb{t}}$ is transposed according to (13) onto frequency space, we loose direct information on its temporal progression of; furthermore, the spectral form is highly abstract in that its "alphabet" consists of an infinite number of eternally sounding "vowels" in the form of the scaled perpetual basis harmonic waves $\exp(i\cdot\varphi t)\big|_{\mathbb{t}}^{\Leftrightarrow}$. It is the entirety of these spectral "letters" that describes $\underline{f(t)}\big|_{\mathbb{t}}$ in the frequency dimension (in other words the concerted sound, i.e. the superposition of the vowels returns $\underline{f(t)}\big|_{\mathbb{t}}$ in its exact shape and at its precise time location; however, the vowels cancel each other exactly to produce utter "silence" for all $t \notin \mathbb{t}$). This kind of abstraction being inherent to $\underline{F_{\mathbb{t}}(\vec{\omega})}\big|_{\boldsymbol{\omega}}$ is something that is often not fully appreciated, and the fact that the FT involves transposition between the physical variables time and frequency invites a strong reflexive (and often unjustified) association of its mathematics with physical properties. In particular, since the "letters" of the spectrum are formed of harmonic waves, one may be inclined to think (in violation of the LCPE) that each of the (abstract) letters $\exp(i\cdot\varphi t)\big|_{\mathbb{t}}^{\Leftrightarrow}$ represents a physically existing oscillation perceived as being somehow inherent in the temporal function but "revealed" by the FT. However, the spectral letters happen to be harmonic waves simply as a consequence of the mathematics of the FT. We can envisage the basis set $\exp(i\cdot\vec{\omega}t)\big|_{\mathbb{t}}^{\Leftrightarrow}$ on which the forward FT transposes $\underline{f(t)}\big|_{\mathbb{t}}$ as an abstract "ocean" of harmonic waves stretching infinitely in both the frequency and time dimensions. The forward FT projects the properties of $\underline{f(t)}\big|_{\mathbb{t}}$

onto this "ocean" by adjusting the amplitudes and phases of the waves according to the "coding instructions" of the FT, and the backward FT "decodes" this information to reconstruct $\underline{f}(t)\big|_{\mathbf{t}}$. However, $\exp(i\cdot\vec{\omega}t)\big|_{\mathbf{t}}^{\Leftrightarrow}$ is just (a rather convenient and useful) mathematical vehicle whose waves act as carriers of the FT codes but do not *ab ovo* represent physical oscillations (in principle it would be possible to "translate" $\underline{f}(t)\big|_{\mathbf{t}}$ into an infinite number of other mathematical "languages" whose letters would be based on functional forms other than $\exp(i\cdot\phi t)\big|_{\mathbf{t}}^{\Leftrightarrow}$, in which case the letters would less readily suggest some association with a physical meaning).

To better understand this problem it can be useful to think of $\underline{f}(t)\big|_{\mathbf{t}}$ simply as a curve detached initially from any physical meaning which is "morphed" by the FT into the "oceanic profile" of $\underline{F}_{\mathbf{t}}(\vec{\omega})\big|_{\boldsymbol{\omega}}$. There are certain features of the temporal curve that we, for convenience, can call by specific names (such as 'oscillation frequency', 'phase', 'amplitude', in the case of, say, a sinusoid). If we render specific physical properties through these features to our temporal signal, then we must assign the same physical properties to the spectral representation according to the LCPE since the FT does not "know" anything about the physics represented by those curves (thus it cannot "create" or "reveal" e.g. any frequency "uncertainty" in the frequency domain that was not there in the time domain). Whatever meaning we therefore attribute to $\underline{F}_{\mathbf{t}}(\vec{\omega})\big|_{\boldsymbol{\omega}}$, that meaning emerges through a process of its "a posteriori" physical interpretation, and this must be in keeping with our physical interpretation of the relevant temporal "curve".

Consider e.g. the simple monochromatic real-valued eternal cosine wave $h_{\varpi_@}(t)\big|_{\mathbf{t}}^{\Leftrightarrow}$ and its spectrum $\underline{H}_{\mathbf{t}}^{\Leftrightarrow}(\vec{\omega})\big|_{\boldsymbol{\omega}_@}^{\uparrow}$ [Figure 3(d)]. In $h_{\varpi_@}(t)\big|_{\mathbf{t}}^{\Leftrightarrow}$ the wave's frequency $\varpi_@$ is known exactly by definition and $\underline{H}_{\mathbf{t}}^{\Leftrightarrow}(\vec{\omega})\big|_{\boldsymbol{\omega}_@}^{\uparrow}$ is localized to $\phi_@^{\pm}$, representing the limit where $\wedge t\text{–}h_{\varpi_@}(t)\big|_{\mathbf{t}}^{\Leftrightarrow} = \infty$ and so $\wedge\phi\text{–}\neg\underline{H}_{\mathbf{t}}^{\Leftrightarrow}(\vec{\omega})\big|_{\boldsymbol{\omega}_@}^{\uparrow} = 0$ in (24). Take, next, the time-windowed harmonic waves $h_{\varpi_@}(t)\big|_{\mathbf{t}}^{\bullet\text{-}\bullet}$ and their spectra $\underline{H}_{\mathbf{t}}^{\bullet\text{-}\bullet}(\vec{\omega}_{\succ\phi_@})\big|_{\boldsymbol{\omega}}^{\Leftrightarrow}$ (Figure 4).

Again, in $h_{\varpi_@}(t)\big|_{\mathbf{t}}^{\bullet\text{-}\bullet}$ the frequency $\varpi_@$ has an exactly known value. Since now $\wedge t\text{–}h_{\varpi_@}(t)\big|_{\mathbf{t}}^{\bullet\text{-}\bullet}$ is finite, the FUP requires that $\wedge\phi\text{–}\neg\underline{H}_{\mathbf{t}}^{\bullet\text{-}\bullet}(\vec{\omega}_{\succ\phi_@})\big|_{\boldsymbol{\omega}}^{\Leftrightarrow}$ must also broaden to some finite value. However, just because we have time-windowed $h_{\varpi_@}(t)\big|_{\mathbf{t}}^{\Leftrightarrow}$ its frequency $\varpi_@$ has not become "uncertain". The fact that $\wedge\phi\text{–}\neg\underline{H}_{\mathbf{t}}^{\bullet\text{-}\bullet}(\vec{\omega}_{\succ\phi_@})\big|_{\boldsymbol{\omega}}^{\Leftrightarrow} > 0$ is not a reflection of any "uncertainty" in $\varpi_@$ but a mathematical necessity ensuing from the fact that all information that specifies the temporal function or signal must be uniquely coded into the spectrum. One way of thinking about this is that while for $h_{\varpi_@}(t)\big|_{\mathbf{t}}^{\Leftrightarrow}$ there was no need for the spectrum to code for the

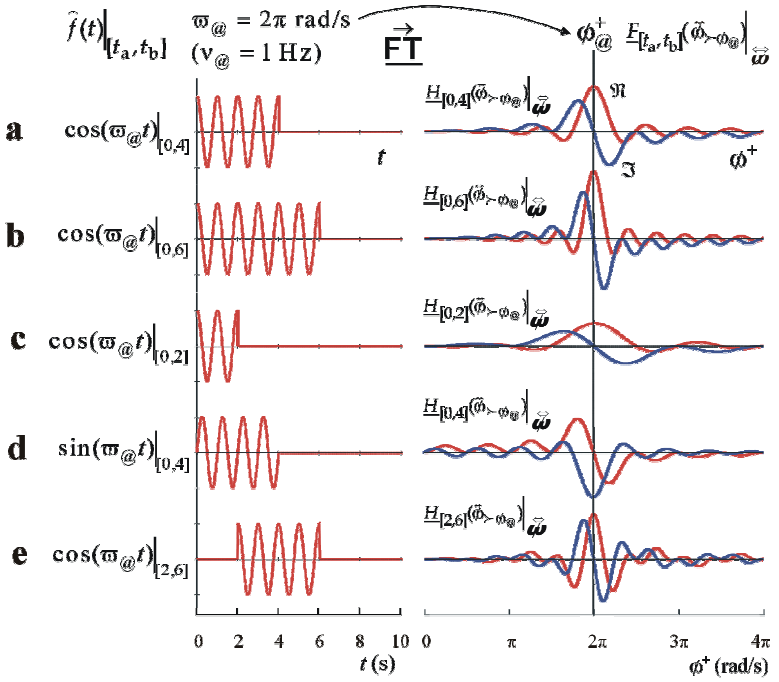


Fig. 4. Simple time-windowed monochromatic temporal waves and their ϕ -space spectra (for simplicity, only the positive phasor-frequency side of the spectra is shown). Each wave's frequency is the same value, $\varpi_@$, but their duration, time location, and initial phase are different. The value of $\varpi_@$ "codes into" the frequency location of the center of the sinc-shaped spectra, therefore this location is the same in each spectrum. Differences in the spectral profiles reflect differences in the waves' other features, because those are coded exactly by the FT into the (complex) FT-spectrum. Spectral broadening is due to that fact that the FT-spectrum carries all that coding information, and not to any "uncertainty" in $\varpi_@$ duration and time location of the wave (which is why the spectrum is a Dirac-delta $\underline{H}_{\overleftrightarrow{t}}(\tilde{\omega})\big|_{\overleftrightarrow{\omega}_@}$), for $h_{\varpi_@}(t)\big|_{\overleftrightarrow{t}}$ a Dirac-delta would not provide enough degrees of freedom in its spectral features to be able to code for the wave's duration and time location. It is, then, this encoding of the "window" into the FT-spectrum that manifests itself in a spectral profile that covers the infinite frequency set $\overleftrightarrow{\omega}$, and not some kind of "uncertainty" in $\varpi_@$. Thinking in terms of "curves", we can render to the curve of $h_{\varpi_@}(t)\big|_{\overleftrightarrow{t}}$ [Figure 4(a)] some distinct physical "aspects", such as a "frequency aspect" defined through the term $\cos(\varpi_@ t)$, and a "window aspect" defined through the location and duration of the operative interval $\overleftrightarrow{t}$. All of these features are coded exactly into the shape of the spectrum $\underline{H}_{\overleftrightarrow{t}}(\tilde{\omega}_{\succ\omega_@})\big|_{\overleftrightarrow{\omega}}$ [Figure 4(a)]. The "frequency aspect" of $h_{\varpi_@}(t)\big|_{\overleftrightarrow{t}}$ translates into the (exact) central

frequency value $\omega_{@}^{\pm}$ of $\underline{H}_{t}^{\bullet\text{-}\bullet}(\vec{\omega}_{\succ\omega_{@}})\Big|_{\omega}^{\Leftrightarrow}$, while the "window aspect" translates into the broadened complex profile of the spectrum. One must therefore understand that the scaled basis elements $\exp(i\cdot\omega t)\Big|_{t}^{\Leftrightarrow}$ in $\underline{H}_{t}^{\bullet\text{-}\bullet}(\vec{\omega}_{\succ\omega_{@}})\Big|_{\omega}^{\Leftrightarrow}$ do not necessarily represent physical vibrations that are actually or "effectively" present in $h_{\varpi_{@}}(t)\Big|_{t}^{\bullet\text{-}\bullet}$. Rather, each eternal basis harmonic wave's (complex) amplitude carries a single piece of the entire set of codes that is needed to reconstruct the original temporal function $h_{\varpi_{@}}(t)\Big|_{t}^{\bullet\text{-}\bullet}$. The physical interpretation that we should, then, correctly assign to $\underline{H}_{t}^{\bullet\text{-}\bullet}(\vec{\omega}_{\succ\omega_{@}})\Big|_{\omega}^{\Leftrightarrow}$ is simply that the frequency of the central basis element of the spectrum corresponds to the monochromatic frequency of $h_{\varpi_{@}}(t)\Big|_{t}^{\bullet\text{-}\bullet}$, and the rest of the basis elements code for all of its other features such as its "window aspect".

The popular notion that the forward FT disseminates a temporal function (signal) into its "frequency components" is incorrect if by "frequency component" one understands a physical vibration contained in, and lasting for the duration of, the signal. Rather, the spectrum represents a transposition of all properties of the temporal signal into frequency space, and some of those properties may happen to be periodic events whose frequency will show up as peaks in the spectrum, but other features of the signal also become scrambled into the spectrum. Claiming therefore that any given point of the FT-spectrum represents a "frequency component" of the temporal signal is misleading and violates the LCPE. Indeed, Figures 4(b-e) illustrate this point by showing that if we keep the monochromatic "frequency aspect" of $h_{\varpi_{@}}(t)\Big|_{t}^{\bullet\text{-}\bullet}$ constant but vary its other "aspects", the spectral profiles will change so as to code for every alteration in the temporal curve, but the (unchanged) value of $\varpi_{@}$ remains "coded into" the same frequency position for each spectrum.

4.3 Back to NMR

If one reconsiders the statements on PFT-NMR quoted in the Introduction, the flaws in those arguments should now be evident. An RF pulse can be treated as a deterministic, time-windowed harmonic phasor in that it acts as a classical magnetic field vector rotating with a specific monochromatic frequency. The assertion that "a nominally monochromatic short pulse effectively has a (Heisenberg) uncertainty-broadened frequency band" simply reflects a misuse of the HUP and/or a physical misinterpretation of the FUP. The FUP does not warrant the causal inference that "wide frequency band" should mean a wide range of physically existing driving RF oscillations, as is often deemed necessary for explaining the phenomenon of off-resonance excitation. Actually, according to the LCPE the sinc-shaped spectral profile of the pulse is synonymous in physical meaning with its temporal form, therefore the statement that a "short pulse has a wide frequency band" does not tell us any more "physics" about the pulse than saying that a "short pulse is a short pulse". In fact there is nothing paradoxical about the fact that a monochromatic pulse can excite off-resonant spins: the phenomenon is solely due to the pulse's large amplitude, akin to the way a classical mechanical resonator also gives nonzero response to a sufficiently large off-

resonance driving force. Nota bene: a) the common notion that "a shorter pulse excites a wider frequency band" is misleading in that it is valid if by 'excites' one specifically means 'gives absorption-shaped measured signals', but is not valid, and thus does not conflict with the above statements, if by 'excites' one just means 'perturbs' in general (Szántay, 2008c); b) the response of a linear system to an input signal can in fact be calculated from the spectrum of the input signal by way of the so-called Superposition Principle (the NMR system is nonlinear in general, and the linearity approximation is valid only for special limiting conditions), but that concept should not be confused with the ideas discussed above.

5. Conclusion

The Fourier Transform (FT) is often thought of as a mathematical device capable of "revealing" the "frequency components" of a temporal signal. Although this notion is correct with regard to any sinusoid oscillatory feature that may be present in the signal, it is deceptive in general, easily leading to the belief that the overall spectral profile represents all the physical "frequency components" that are present in the signal. This idea can give rise to false interpretations such as the concept that the frequency of a time-limited monochromatic sinusoid is "uncertain" or "effectively polychromatic", in line with the "Uncertainty Principle". In fact the Heisenberg Uncertainty Principle (HUP) has nothing to do with this problem; it is the Fourier Uncertainty Principle (FUP) that is of relevance, but the word "uncertainty" is also misleading in that regard: the FUP does not imply that time-windowing a deterministic wave with a monochromatic frequency makes its frequency "uncertain" or that the forward FT "reveals" such uncertainty just because the spectrum has a profile that covers a broad range of frequencies. Whatever physical attributes we associate with a temporal signal (such as that it has an exact monochromatic frequency) must be carried over exactly into the frequency dimension according to the "Law of Conjugate Physical Equivalence" (LCPE).

6. Acknowledgement

The author is indebted to Dr. Zsuzsanna Sánta for her critical and constructive mathematical insights.

7. References

Bloch, F. (1946). Nuclear Induction, *Physical Reviews*, Vol. 70, pp. 460-474.

Claerbout, J. F. (2004). *Earth Sounding Analysis: Processing versus Inversion*. pp. 256-259. Available at: http://sep.stanford.edu/sep/prof/pvi.pdf

Cohen, L. (1995). *Time-frequency analysis*. Prentice Hall, New Jersey, pp. 44-46 and pp. 195-197.

Derome, A. E. (1987). *Modern NMR Techniques for Chemistry Research*. Pergamon, New York, p. 12.

Fukushima E., & Roeder, S. B. W. (1981). *Experimental Pulse NMR, A Nuts and Bolts Approach*. Addison-Wesley, London, p. 51.

Hanson, L. G. (2008). Is Quantum Mechanics Necessary for Understanding Magnetic Resonance? *Concepts in Magnetic Resonance*, Vol.32, pp. 329-340.

Harris, R. K. (1983). *Nuclear Magnetic Resonance Spectroscopy, a Physicochemical View*. Pitman, London, p. 74.

Hoult, D. I. (1989). The Magnetic Resonance Myth of Radio Waves. *Concepts in Magnetic Resonance*, Vol.1, pp. 1-5.

King R. W. & Williams K. R. (1989). The Fourier Transform in Chemistry. *Journal of Chemical Education*, Vol.66, pp. A213-A219.

Marshall A. G. & Verdun, F. R. (1990). *Fourier Transforms in NMR, Optical, and Mass Spectrometry,* Elsevier, New York, p. 31.

Szántay, C., Jr. (2007). NMR and the Uncertainty Principle: How to and How not to Interpret Homogeneous Line Broadening and Pulse Nonselectivity. Part I. The Fundemantals. *Concepts in Magnetic Resonance,* Vol.30A, pp. 309-348.

Szántay, C., Jr. (2008a). NMR and the Uncertainty Principle: How to and How not to Interpret Homogeneous Line Broadening and Pulse Nonselectivity. Part II. The Fourier Connection. *Concepts in Magnetic Resonance,* Vol.32A, pp. 1-33.

Szántay, C., Jr. (2008b). NMR and the Uncertainty Principle: How to and How not to Interpret Homogeneous Line Broadening and Pulse Nonselectivity. Part III. Uncertainty? *Concepts in Magnetic Resonance,* Vol.32A, pp. 302-325.

Szántay, C., Jr. (2008c). NMR and the Uncertainty Principle: How to and How not to Interpret Homogeneous Line Broadening and Pulse Nonselectivity. Part IV. Un(?)certainty. *Concepts in Magnetic Resonance,* Vol.32A, pp. 373-404.

10

Aspects of Using Chirp Excitation for Estimation of Bioimpedance Spectrum

Toivo Paavle, Mart Min and Toomas Parve
Th. J. Seebeck Dept. of Electronics
Tallinn University of Technology,
Estonia

1. Introduction

Short frequency swept signals, known as chirps, are widely used as excitation or stimulus signals in various areas of engineering as radar and sonar techniques, acoustics and ultrasonics, optical and seismological studies, but also in biomedical investigations, including bioimpedance measurement and impedance spectroscopy (Müller & Massarani, 2001; Misaridis & Jensen, 2005; Barsoukov & Macdonald, 2005; Nahvi & Hoyle, 2009).

Signal processing in the chirp-based applications is often combined with pulse compression via cross-correlation procedure and Fourier analysis. In this chapter, a similar approach is proposed for estimation of the frequency response (the impedance spectrum) of electrical bioimpedance. An advantage of the chirp-based method is that the characteristics of a biological object can be obtained in a wide frequency range during a very short measurement cycle, which nearly eliminates the influence of low-frequency biological processes (heart beating, breathing, pulsation of blood) to the result of measurement.

The changes of a spectrum monitored at sequent time intervals by means of the Fourier Transform (known as spectrogram) is an informative base for interpreting the processes in biological objects. Furthermore, the signal treatment by the cross-correlation yields a better noise immunity for the measurement system (Barsoukov & Macdonald, 2005), and adds some alternatives for estimation of the bio-impedance properties and behavior.

2. Basics of bioimpedance measurement

2.1 Object of measurement

The impedance of living tissues or, in general, of arbitrary biological matter (electrical bioimpedance, EBI) can be characterized by its electrical equivalent, which, in turn, can be represented as the frequency-dependent complex vector $\dot{Z}(j\omega) = \mathrm{Re}(\dot{Z}(j\omega)) + j\mathrm{Im}(\dot{Z}(j\omega)) = Z(\omega)\exp(j\Phi_z(\omega))$, where $\omega = 2\pi f$, $Z(\omega) = (\mathrm{Re}(\dot{Z}(j\omega))^2 + \mathrm{Im}(\dot{Z}(j\omega))^2)^{1/2}$, and $\Phi_z(\omega) = \mathrm{arctg}(\mathrm{Im}(\dot{Z}(j\omega))/\mathrm{Re}(\dot{Z}(j\omega)))$.

On the other hand, the bioimpedance is not a constant value, but a function, which is changing in time due to numerous biological processes in the living tissue, and as a matter of fact, is a function of frequency and time altogether as $\dot{Z}=Z(j\omega,t)$.

Injection of the excitation current (stimulus) $I_{exc}(t)$ with known parameters into biological object courses the response voltage $V_z(t)$, analysis of which enables to estimate the impedance spectrum $\dot{Z}(\omega)$ of the object being under the investigation.

Fig. 1 illustrates the path of excitation current through the cells of a tissue, where r_{ext} corresponds to the extracellular resistance, and the resistive components r_{int} together with intercellular capacitances C_c constitutes the intracellular impedance (Grimnes & Martinsen, 2008; Min & Parve, 2007).

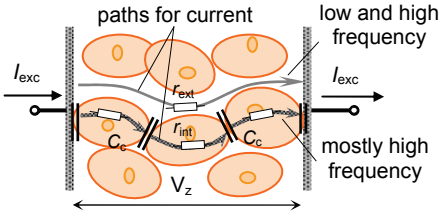


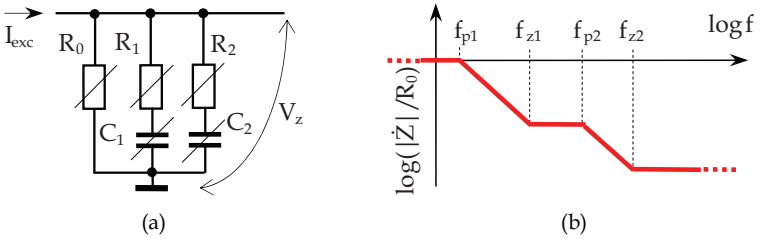
Fig. 1. Formation of the electrical bioimpedance of tissue

Very often, the response signal is analyzed using the Fourier Transform $\boldsymbol{F}(V_z(t))$ to get information about the frequency-dependent state and changes of the biological matter. Here, the impedance spectrum of EBI manifests as $\dot{Z}(j\omega)=\boldsymbol{F}(V_z(t))/\boldsymbol{F}(I_{exc}(t))$, i.e., the Fourier Transform of the response voltage determines the impedance spectrum of the object one-to-one thanks to the predetermined parameters of the excitation signal.

Usually, in theoretical considerations and simulations, the bioimpedance is substituted by a certain RC-circuit. Naturally, the accuracy of such approximation depends on the number and configuration of components. In Fig. 2a, a 5-element circuit is shown, where R_0 corresponds to the extracellular resistance r_{ext} of a tissue, while R_1, C_1, R_2 and C_2 stand for the intracellular parameters. The respective Bode diagram has two real poles f_{p1}, f_{p2}, and two zeros f_{z1}, f_{z2} (Fig. 2b), spread typically over the frequency range from some kHz up to several MHz (Nebuya *et al* 1999; Pliquett *et al*, 2000; Grimnes & Martinsen, 2008).

The Laplace transform of the 5-element EBI (Fig. 2a) can be expressed as (Paavle *et al* 2008)

$$Z(s)=\frac{R_0(1+sC_1R_1)(1+sC_2R_2)}{R_0\left[sC_2(1+sC_1R_1)+sC_1(1+sC_2R_2)\right]+(1+sC_1R_1)(1+sC_2R_2)} \tag{1}$$



(a) (b)

Fig. 2. (a) 5-element model and (b) Bode diagram of the bioimpedance

In the simplest case, where $C_2 \to 0$ (basic 3-element EBI):

$$Z(s) = \frac{R_0(1 + sC_1R_1)}{1 + sC_1(R_0 + R_1)} \tag{2}$$

For adequate estimation of the bioimpedance, the spectrum of excitation signal should cover the frequency range of the object $\dot{Z}$ as much as possible. For that reason, several types of broadband excitation signals (e.g., Maximal Length Sequence (Gawad *et al*, 2007), multisine (Sanchez *et al* 2011)) are preferable. However, in this chapter, we will focus on the use of chirp excitation due to several advantages of it: wide and flat amplitude spectrum (Nahvi & Hoyle, 2009) together with independent scalability in the time and frequency domain.

2.2 Essential of chirp excitation

2.2.1 Variety of chirps

A sine-wave based chirp with current phase $\theta(t)$ can be described mathematically as

$$V_{\mathrm{ch}}(t) = A\sin\left(\theta(t)\right) = A\sin\left(2\pi\int f(t)\mathrm{d}t\right) \tag{3}$$

where A is the amplitude, and $f(t) = (\mathrm{d}\theta(t)/\mathrm{d}t)/2\pi$ is instantaneous frequency of the chirp signal.

A definite class of chirps has the instantaneous frequency $f(t)$, which changes accordingly to some power function of the n^{th} order (power chirps). Their specific quantity is chirping rate $\beta=(f_{\mathrm{fin}}-f_{\mathrm{st}})/T_{\mathrm{ch}}^{n}$, where f_{st} and f_{fin} are the initial and final frequencies, respectively, and T_{ch} is the duration of the pulse. Chirps of this type can be expressed as

$$V_{\mathrm{ch}}(t) = A\sin\left(2\pi\left(f_{\mathrm{st}}t + \beta t^{n+1}/(n+1)\right)\right) \tag{4}$$

The instantaneous frequency of a chirp can be increasing (up-chirps, $\beta>0$) or decreasing (down-chirps, $\beta<0$) quantity. Besides, sometimes it is practical to generate chirps with a symmetrical bidirectional frequency change (bidirectional or double chirps). In this case, the actual duration of the pulse is $2T_{\mathrm{ch}}$, and the sign of chirping rate alters at $t = T_{\mathrm{ch}}$, causing mirrored waveform of the pulse against that moment.

Fig.3 sketches waveforms of different chirps with the equal pulse duration and almost equal frequency range. A very basic linear (n=1) chirp with f_{st}=0 can be described in accordance with the expression (4) as $V_{\mathrm{ch}}(t)=A\sin(2\pi f_{\mathrm{fin}}t^2/2T_{\mathrm{ch}})$. It is depicted in Fig. 3a.

Fig. 3b shows the waveform of quadratic (n=2) down-chirp. However, the rule of frequency change can be arbitrary. For example, in some specific measurements, excitation with the exponential chirping rate $\beta = (f_{\mathrm{fin}}/f_{\mathrm{st}})^{t/T\mathrm{ch}}$ is appropriate (Darowicki & Slepski, 2004). The waveform of sinusoidal exponential chirps (see Fig. 3c) is described as

$$V_{\mathrm{ch}}(t) = \sin\left(2\pi f_{st}T_{\mathrm{ch}}(\beta - 1)/\ln(\beta)\right) \tag{5}$$

Generation of a perfect sine-wave chirp requires quite complicated hardware, which can cause problems, especially in on-chip solutions. That is why so-called signum-chirps (known also as pseudo-, binary- or Non-Return-to-Zero (NRZ) chirps) are often implemented (Figs. 3d and 3e; the latter one depicts a binary chirp with bidirectional change

of frequency). This kind of chirps can be defined by the signum-function of respective sine-wave chirps as $V_{ch}(t)=V_{sgn}(t)=\text{sign}(V_{sin}(t))$, which have binary values $+A$ and $-A$ only.

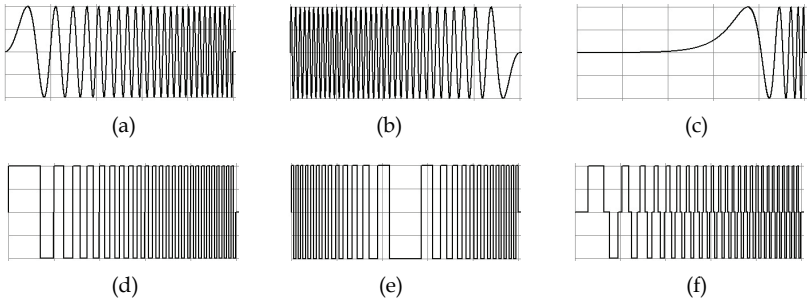


Fig. 3. Examples of chirp waveforms with the equal maximal frequency and duration, $V_{ch}(t)$ vs. time: (a) linear sine-wave chirp; (b) quadratic down-chirp; (c) exponential chirp; (d) signum-chirp; (e) signum-chirp with bidirectional (down-up) run of frequency; (f) ternary chirp with shortened duty cycle by 30°

Rectangular waveforms simplify signal processing: both generation of excitation and processing of the response will be substantially simpler. Especially simple is calculating of the correlation function or deconvolution in the time domain – shifting and multiplication with a reference signal having only {+1, -1}, or {+1, 0, -1} values (Rufer *et al*, 2005).

Additional advantage of signum-chirps is their unity crest factor (i.e., the peak amplitude ratio to the root-mean-square (RMS) value of the signal), and major energy compared with the sine-wave chirps of the same length. Unfortunately, using of rectangular signals causes the worse purity of the spectrum due to the accompanying higher harmonic components.

Suppression of the particular harmonic component in the spectrum of rectangular signals can be achieved by shortening the duty cycle of the signal by a certain degree α_d per every quarter-period (Parve & Land, 2004). Signum-chirps, modified in this way, are called Return-to-Zero (RZ) or ternary chirps. It means that $V_{sgn}(t)$ returns to zero, if the value of current phase falls into the intervals $2n\pi-\alpha_r<\theta(t)<2n\pi+\alpha_r$ or $(2n+1)\pi-\alpha_r<\theta(t)<(2n+1)\pi+\alpha_r$, where $\alpha_r=\pi\alpha_d/180$ in radians, and n=0, 1, 2, … (Fig. 3f). For explanation let us remember that spectra of a rectangular signal can be declared as the Fourier series of odd harmonics:

$$\boldsymbol{F}(\omega t)=\frac{4A}{\pi}\left[\frac{\cos\alpha_r}{1}\sin\omega t+\frac{\cos 3\alpha_r}{3}\sin 3\omega t+\ldots\right]=\frac{4A}{\pi}\left[\sum_{i=1}^{\infty}\frac{\cos(2i-1)\alpha_r}{2i-1}\sin(2i-1)\omega t\right] \quad (6)$$

It follows from (6) that the k^{th} harmonic (k=2i–1, and i=1, 2, 3, …) is absent from the series $\boldsymbol{F}(\omega t)$, when $k\alpha_r=\pm(2n+1)\pi/2$, and n=0, 1, 2, … , because of $\cos(k\alpha_r)=0$. Consequently, in the case of $\alpha_r=\pi/6$, the $(3+6n)^{th}$ harmonics are removed, and for $\alpha_r=\pi/10$, the $(5+10n)^{th}$ harmonics are removed, etc. (Paavle et al, 2007).

2.2.2 Short-time chirps (titlets)

Commonly, referring to chirps, signals of many cycles (rotations by 2π of the chirp generating vector) are considered (multi-cycle chirps). It does not need to be so, and the

chirps with a single cycle or even less ($\theta(T_{ch}) \le 2\pi$) can be generated and used, too. For such kind of ultra short chirps, the neologism "titlets" or Minimal Length Chirps (MLC) have been used (Min et al, 2011a).

The titlets can be very effective excitation signals in applications, where broadband excitation is necessary, but the minimal power consumption or extremely fast measurement are required at the same time. Because of the prospective use of titlets, a special attention to their properties will be paid below. The diagram in Fig. 4 explains the forming of a single-cycle linear sine-wave chirp.

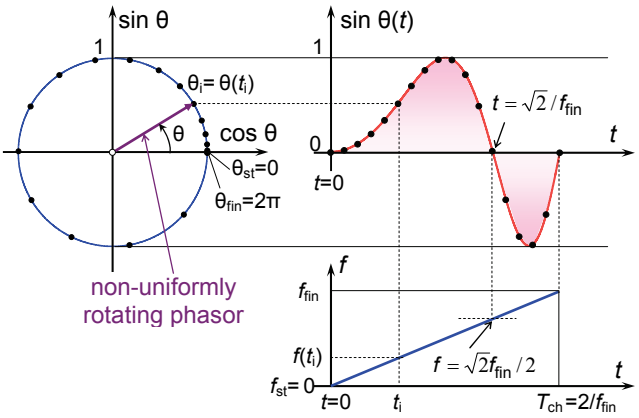


Fig. 4. Genesis of the linear single-cycle sine-wave chirp

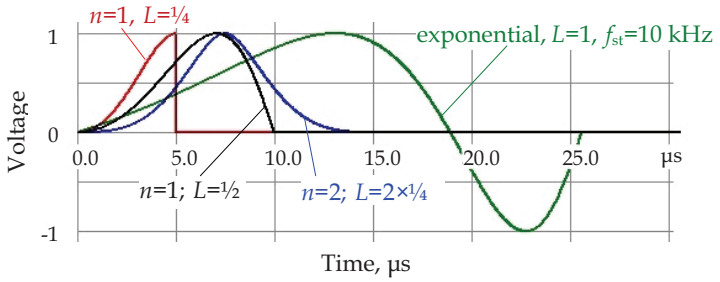


Fig. 5. Examples of titlet waveforms with f_{fin}=100 kHz (f_{st}=0 for power titlets)

Considering (4), we can state that the equality of phases $\theta(t) = 2\pi(f_{st}T_{ch}+\beta T_{ch}/(n+1) = 2\pi L$ is valid for the $t = T_{ch}$. Thereby, the number of cycles L can be an integer or fractional quantity. It allows us to derive the relationship between the length of the chirp and its parameters. For the n^{th} order power chirps this relationship expresses as (Min *et al*, 2011a)

$$T_{ch} = L(n+1)/(nf_{st} + f_{fin}) \tag{7}$$

Thus, for the single-cycle linear chirp with f_{st}=0, the pulse duration T_{ch}=2/f_{fin}. It follows for this case that the change of polarity ($\theta(t) = \pi$) occurs at $t = \sqrt{2} / f_{fin}$, while the instantaneous frequency is $f = \sqrt{2} f_{fin} / 2$ (see Fig. 4).

Similarly, considering (5), one can show that for exponential chirps the duration of a pulse is

$$T_{ch} = L \cdot \ln\left(f_{fin} / f_{st}\right) / \left(f_{fin} - f_{st}\right) \tag{8}$$

Simulated waveforms of some typical titlets with $L \leq 1$ are shown in Fig. 5.

2.3 Signal analysis and bioimpedance measurement by using Fourier Transform

Spectral analysis is the irreplaceable method as for the study of the behavior of excitation signals in the frequency domain as well as for processing the response signal in the bioimpedance measurements.

It is obligatory to remember that in the modeling and processing of chirp signals the selected sampling rate f_s must follow the Nyqvist criterion over the whole frequency range, i.e., the condition $f_s \geq 2f_{fin}$ must be fulfilled. The sampling rate determines the total number of samples N_{ch} during a chirped pulse as $N_{ch} = T_{ch} f_s$. Yet, the simulation or real processing time T_{tot} can be longer than T_{ch}. In this case, the zero padding is used at $t > T_{ch}$. The length of the input array for the Fast Fourier Transform (FFT) is $N = T_{tot} f_s \geq N_{ch}$, and it determines the frequency resolution Δf (difference between two successive frequency bins) of the FFT processing, whereas $\Delta f = f_s / N$ according to the uncertainty principle (Vaseghi, 2006). The acquired positive frequency range of the FFT processing in Hz is 0, …, $N\Delta f/2$ by steps of Δf.

2.3.1 Principles of bioimpedance measurements

There are several methods to calculate the frequency spectrum of the bioimpedance response. A typical way is the implementation of two FFT-channels for the response and excitation signals separately (Min *et al*, 2011a). Nevertheless, in the following discussion we will focus mainly on the structure and modeling of a specific bioimpedance measurement system, which incorporates the cross-correlation procedure together with following FFT-processing (Paavle *et al*, 2008, Min *et al*, 2009, see Fig. 6).

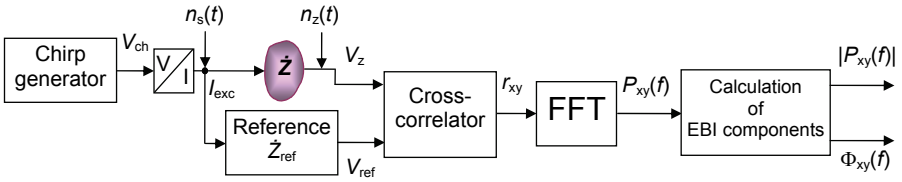


Fig. 6. Basic structure of the measurement system

In such a system, the cross-correlation function (CCF) is calculated: (a) between the response $V_z(t)$ and excitation $V_{ch}(t)$ at the unity gain in the reference channel ($V_{ref}(t) = V_{ch}(t)$, $\dot{Z}_{ref} = 1$), or (b) between the response and predefined reference signal $V_{ref}(t)$ (reference channel includes the known impedance $\dot{Z}_{ref} \neq 1$). In the latter case, the system works as the matched filter, enabling detection of mismatches between the impedance vectors $\dot{Z}$ and $\dot{Z}_{ref}$ more precisely together with somewhat better noise reduction. The source noise $n_s(t)$ and object noise $n_z(t)$ can be taken into account in simulations as shown in Fig. 6.

In general, calculation of the CCF proceeds as

$$r_{xy}(\tau) = Corr\left(V_z(t), V_{ref}(t)\right) = \overline{V_z(t)V_{ref}(t+\tau)}, \tag{9}$$

where τ is a variable delay (lag) and the overline denotes averaging.

Using broadband chirp excitation, the waveform $r_{xy}(\tau)$ of CCF is similar to the $\sin(x)/x$ type sinc-function affected by the nature of $V_z(t)$ and is strongly compressed close to $\tau = 0$.

The CCF includes both the information about the amplitude level and phase shift of the bioimpedance vector. Thanks to this fact, only a single FFT-block is necessary to obtain the complete information about the object. Otherwise, when using the direct Fourier Transform of the response, another FFT channel is required to establish the basis for phase evaluations.

In accordance with the cross-correlation theorem (generalized Wiener-Khinchin theorem), the correlation function is equivalent to the inverse Fourier Transform of the complex cross-power spectrum density $P_{xy}(f)$ with the magnitude and phase spectral components $\left|P_{xy}(f)\right| = \sqrt{(\mathrm{Re}(P_{xy}(f)))^2 + (\mathrm{Im}(P_{xy}(f)))^2}$ and $\Phi_{xy}(f)=\mathrm{arctg}(\mathrm{Im}(P_{xy}(f))/\ \mathrm{Re}(P_{xy}(f)))$, respectively (Vaseghi, 2006). Thereby, the phase component $\Phi_{xy}(f)$ specifies the phase difference between the vectors V_z and V_{ref}, and magnitude presents the geometric mean of the power spectral densities of the signals $V_z(t)$ and $V_{ref}(t)$ (McGhee *et al*, 2001).

2.3.2 Modeling of the measurement system

Accordingly to the main architecture of measurement system (Fig. 6) and model of EBI (Fig. 2a), a special PC-model (Fig. 7) was developed for verifying the theoretical conceptions (Paavle *et al*, 2008).

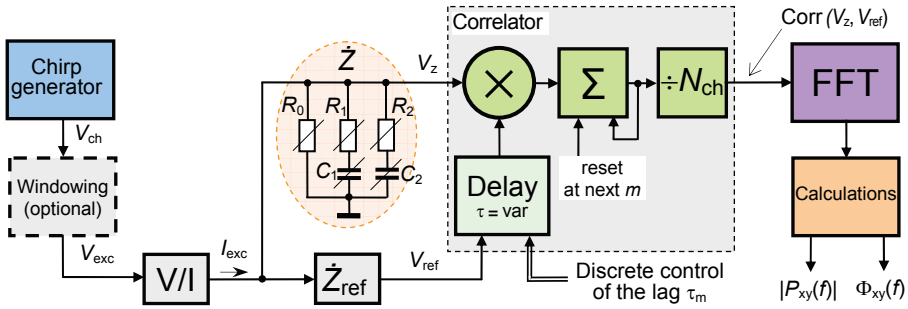


Fig. 7. Model of the measurement system

The cross-correlation function is calculated by the cumulative adder for every m^{th} lag τ_m as

$$r_{xy}[m] = \frac{1}{N_{ch}} \sum_{k=0}^{N_{ch}-1} V_z[k] \cdot V_{ref}[k+\tau_m], \tag{10}$$

where $k = 0 \ldots N_{ch}-1$, and N_{ch} is the number of samples per length of the chirp pulse (McGhee *et al*, 2001).

If the uniform delay step is $\Delta\tau$, then for the overall delay interval (observation time) $t_{obs} = \tau_{max} - \tau_{min}$ ($\tau_{min} \le \tau_m \le \tau_{max}$) the number of necessary computing cycles is $M + 1$ with $M = t_{obs}/\Delta\tau$. As a result, we acquire an array of correlation values $r_{xy}[m]$ with $m = 0 \dots M$. Supposing that the selected $\Delta\tau$ and τ_{min} are integer multiples of the sampling interval $1/f_s$, the m^{th} delay expresses as $\tau_m = f_s(\tau_{min} + m\Delta\tau)$.

In the following Fourier Transform, the array r_{xy} of length $M+1$ represents the input stream for the FFT-block, while in accordance with the uncertainty principle (see above), the frequency resolution of spectra becomes $\Delta f = 1/t_{obs}$ over the frequency range $f = 0 \dots 1/(2\Delta\tau)$.

3. Spectral features and energy of chirps

Next, let us take a look at the results of direct Fourier Transform of distinct chirp pulses. The direct FFT of different excitation signals enables to compare their amplitude and power spectra together with estimation of the energy properties keeping in view the requirements for the bioimpedance measurement.

Important and desired spectral properties of excitation signals to improve the quality of wideband measurement are:

- flat amplitude spectrum with minimal fluctuation (ripple) together with the absence of overshoots inside the generated (excitation) bandwidth $B_{exc} = f_{fin} - f_{st}$;
- steep drop-down of the amplitude spectrum outside the bandwidth B_{exc};
- maximal energy-efficiency, i.e., the ratio between the energy lying within the generated (excitation) bandwidth B_{exc} and total energy of the signal.

For some specific applications, these properties must coincide with minor power consumption and with shortness of signal to ensure quick measurement.

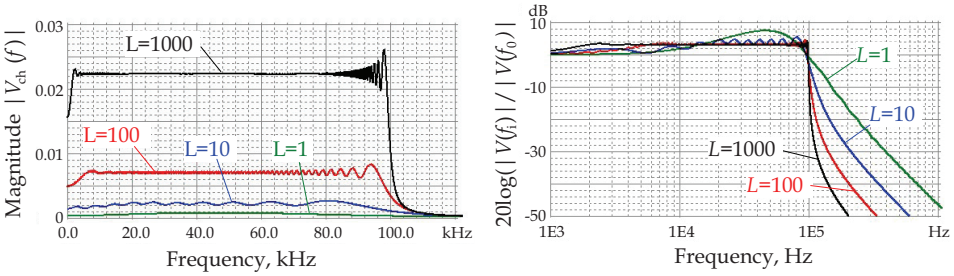


Fig. 8. Voltage spectral density of linear chirps 0…100 kHz (Δf=50 Hz): (a) linear scaling at the unity amplitude of chirp pulses; (b) normalized logarithmic scaling

Fig. 8 depicts the amplitude spectra of linear sine-wave chirps of different length in the linear and in the normalized logarithmic scales. Thereby, the base of normalizing is the amplitude $|V(f_0)|$, i.e., the magnitude at the lowest frequency bin f_0 of the Fourier Transform. It is obvious that long chirps assure the more fitting shape of spectra, except a certain rippling caused by the Gibbs effect. The latter can be suppressed by using a kind of windowing in the time domain (often a boxcar-type window function is used for this purpose), but in this case, the total energy of the signal decreases.

Fig. 9 shows the frequency run and respective amplitude spectra of different titlets. It appears that the desired shape of the spectrum, including flatness, satisfying drop-down outside the bandwidth (slopes from –20 dB/dec to –80 dB/dec were observed) and admissible overshoots can be achieved by a proper selection of the type and length of titlet. Moreover, additional correction and shaping of spectra can be attained by windowing of titlets in the time domain (see below in Sect. 3.2). In Fig.10, waveforms and amplitude spectra of some rectangular chirps (see Sect. 2.2.1) with L=10 are shown. Naturally, for this kind of chirps, the rippling of spectra is noticeable.

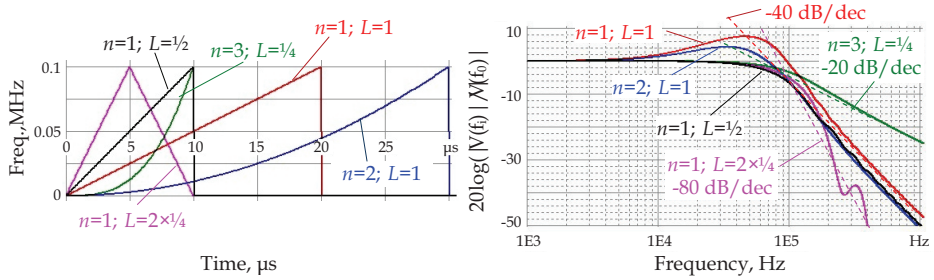


Fig. 9. (a) Frequency run and (b) amplitude spectra of different titlets with B_{exc}=0…100 kHz

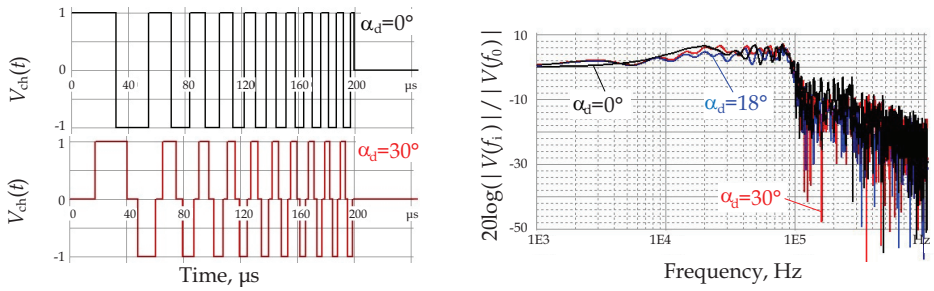


Fig. 10. (a) Waveforms of binary and ternary chirps with α_d=30°; (b) amplitude spectra of binary and ternary chirps with α_d =30° and α_d =18° (Δf=100 Hz)

3.1 Energy of chirps

3.1.1 Sine-wave chirps

In principle, the total energy of a chirp signal at the unity load (R_{load}=1Ω) expresses in time domain as

$$E_{tot} = \int_0^{T_{ch}} V_{ch}(t)^2 dt \,, \tag{11}$$

which leads to $E_{tot} = (A^2/2)T_{ch}$ for the long term chirps of sinusoidal waveform.

According to the Parseval's theorem, the total energy in the frequency domain is the same as the energy in the time domain (Vaseghi, 2006). For chirps with the finite length, a certain

part of signal energy falls outside the useful (generated) bandwidth caused by higher frequency components. For that reason, it is necessary to distinguish the total energy of a generated chirp pulse, which varies proportionally with T_{ch}, and the useful energy E_{exc}, falling inside the chirp bandwidth B_{exc}. Typically, $E_{exc} < E_{tot}$, while absolute values of both quantities depend on the chirp length, waveform and spectral nature. To characterize the percentage of the useful energy, we employ the term of energy-efficiency as $\delta_E = E_{exc}/E_{tot}$. For sine-wave chirps with $L \to \infty$ the ratio $\delta_E \to 1$.

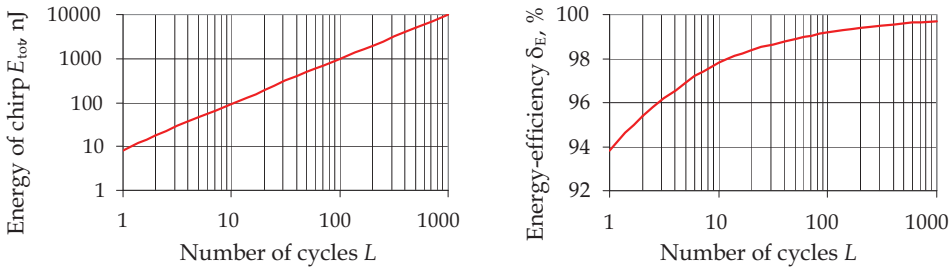


Fig. 11. (a) Total energy and (b) energy-efficiency of linear sine-wave chirps

Type of chirp (titlet)	L	f_{st}, Hz	T_{ch}, μs	P_{avg}, mW	E_{tot}, nJ	δ_E, %
linear	¼	0	5.0	0.32	1.6	55.0
	½	0	10.0	0.38	3.8	90.6
	1	0	20.0	0.41	8.3	93.5
	10	0	200.0	0.47	94.4	97.8
	100	0	2000	0.49	982	99.3
quadratic	½	0	15.0	0.29	4.4	93.9
	1	0	30.0	0.33	10.0	95.7
	100	0	3000	0.464	1393	99.5
cubic	1	0	40.0	0.28	11.1	96.4
exponential	1	1	115.1	0.135	15.6	97.8
	10	1	1151	0.235	271	99.5
double-quadratic	2×¼	0	2×7.5	0.23	3.4	93.1
NRZ signum-chirp	1	0	20.0	1.0	20.0	84.1
	1000	0	20e3	1.0	20e3	85.1
RZ chirp (18° short.)	1000	0	20e3	0.8	16.0e3	93.1
RZ chirp (30° short.)	1000	0	20e3	0.67	13.3e3	92.1

Table 1. Energy and average power of different chirp pulses with $f_{st} \approx 0$ and f_{fin}=100 kHz

Considering expressions (11) and (4) or (5), the analytical description of the chirp energy becomes very complicated. So it is reasonable to calculate the respective quantities numerically. Nevertheless, a good approximation of total energy and energy-efficiency of arbitrary chirp signal can be obtained using the results from the Fourier Transform as follows (Paavle *et al*, 2010):

$$\delta_E = \sum_{i=N_{st}}^{N_{fin}} |V_{ch}(f_i)|^2 \Big/ \sum_{i=0}^{N_{max}-1} |V_{ch}(f_i)|^2 \tag{12}$$

where $|V(f_i)|$ is the value of the amplitude spectrum at i^{th} frequency bin, N_{st} and N_{fin} are the numbers of frequency bins, corresponding to the f_{st} and f_{fin}, respectively. N_{max} is the total number of frequency bins, and the divisor in (12) corresponds to the total energy E_{tot}. Of course, this method enables to calculate the partial energy for any frequency interval inside the range of 0 to $N_{max}\Delta f$.

The curves in Fig. 11 present the dependence of the energy and energy-efficiency of linear sine-wave chirps on the number of chirp cycles graphically. A selection of power and energy parameters for different chirps and titlets, obtained from the FFT-processing by using the expression (12), are converged into Table 1, where the average power $P_{avg} = \frac{1}{T_{ch}} \int_0^{T_{ch}} V_{ch}(t)^2 dt = E_{tot} / T_{ch}$ presumes the 1 kΩ load.

3.1.2 Binary signum-chirps

A specific feature of signum-chirps is the gradually decreasing amplitude spectrum by step width of $2B_{exc}$ as shown in Fig. 12 (Min *et al*, 2009). Let us analyze this phenomenon.

The fundamental harmonic of a regular rectangular signal with amplitude $A_1 = (4/\pi)\,A$ has a root-mean-square (RMS) value $A_1/\sqrt{2} = 4A/(\pi\sqrt{2})$, energy $E_1 = (A_1^2/2)\,T_{ch} = (8/\pi^2)\,A^2\,T_{ch}$, and power $W_1 = E_1/T_{ch} = (8/\pi^2)\,A^2$, which creates a constant value power spectral density (PSD) $w_1 = W_1/B_{exc}$, V²/Hz, within the bandwidth B_{exc} of fundamental harmonic:

$$w_1 = \left(8/\pi^2\right) A^2 / B_{exc} \tag{13}$$

The amplitudes of k^{th} higher harmonics are of $A_k = A_1/k$. Therefore, the power of every k^{th} higher odd harmonic (k =3, 5, 7, …) is equal to W_1/k^2, being spread over the frequency range $B_k = kf_{fin} - kf_{st} = kB_{exc}$. Due to the fact that higher harmonics have k times wider bandwidth than the fundamental (first) harmonic has, the PSD for higher harmonics can be expressed through (12) as $w_k = (w_1/k^2)/k = w_1/k^3$ (Min *et al*, 2009).

The total power of generated signal $V_{ch}(t)$ is gradually distributed over its whole frequency range, theoretically from f_{st} to ∞, see Fig. 12. The PSD p_h of every gradual h^{th} level (h= 1, 2, 3, 4, …) of the spectrum, beginning from the first one p_1, is the sum of power spectral densities of the fundamental and higher harmonics, w_1 and w_k, which contribute into the given level h. Into the PSD p_1 of the first level contribute all the signal components $p_1 = w_1 + w_3 + w_5 + w_7 + \ldots$, but into the second level mere the higher harmonics $p_2 = w_3 + w_5 + w_7 + \ldots$, ($k$ = 3, 5, 7, …), and into the third level only the harmonics beginning from k = 5, $p_3 = w_5 + w_7 + \ldots$, etc.

Generally,

$$p_h \approx w_1 \sum_{i=h}^{h+m} (2i-1)^{-3} \tag{14}$$

in which i is an integer beginning from h, that is: $i = h$, (h+1), (h+2), (h+3), …, (h+m), where m is the number of higher odd harmonics taken into account.

The partial power $P_1 = p_1 B_{exc} = P_{exc}$ of the first level (h=1) is the useful excitation power

$$P_{exc} \approx W_1 \sum_{i=h}^{1+m} (2i-1)^{-3} \tag{15}$$

The power P_{out} falling outside B_{exc} is lost. The lost power can be found as a summed up power of higher harmonics within the next levels of spectrum (h= 2, 3, 4, etc., see Fig. 10):

$$P_{out} \approx W_1 \sum_{i=1}^{1+m} 2(i-1)/(2i-1)^3 \tag{16}$$

where $W_1 = w_1 B_{exc}$ in (15) and (16) is the power of fundamental harmonic. As the minimal frequency for every level is kf_{st}, then the equations (14-16) are approximate ones, which become absolutely exact only for f_{st}=0. In practice, when the excitation bandwidth B_{exc} exceeds one decade significantly, e.g., $f_{fin} / f_{st} > 30$, these equations are exact enough for engineering calculations.

The total chirp energy is $E_{tot} = E_{exc} + E_{out} = (P_{exc} + P_{out})\, T_{ch}$ and the useful excitation energy in it is $E_{exc} = P_{exc}\, T_{ch}$. From (15) and (16) follows the role of useful energy $\delta_E = E_{exc} / E_{tot}$. Considering f_{st}=0, the ratio for the energy-efficiency expresses as

$$\delta_E = \left(\sum_{i=1}^{\max h} (2i-1)^{-3} \right) / \left(\sum_{i=1}^{\max h} (2i-1)^{-2} \right) \tag{17}$$

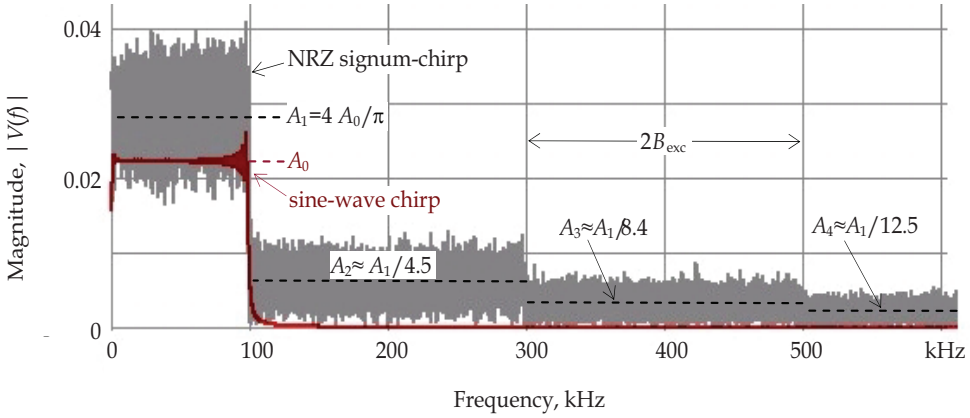


Fig. 12. Amplitude spectra of 1000-cycles sine-wave chirps and signum-chirps with f_{st}=0 and f_{fin}=100 kHz (A=1)

Taking into account all the higher harmonics ($m \to \infty$ and $(\max h) \to \infty$), the sums in (17) will obtain limit values, which can be found via Riemann zeta function (Dwight, 1961):

$$\zeta(x) = \sum_{n=1}^{\infty} n^{-x} \tag{18}$$

Riemann's mathematics gives us the following limit values for the sums in (17):

$$\sum_{i=1}^{\infty}(2i-1)^{-2}=\pi^2/8\,, \tag{19}$$

$$\sum_{i=1}^{\infty}(2i-1)^{-3}=7\zeta(3)/8\,, \tag{20}$$

where $\zeta(3)\approx1.202$ is termed the Apéry's constant (Dwight, 1961). Now, using equations (19) and (20), we can express the part of useful energy in the linear NRZ signum-chirp pulse as

$$\delta_E=7\zeta(3)/\pi^2 \tag{21}$$

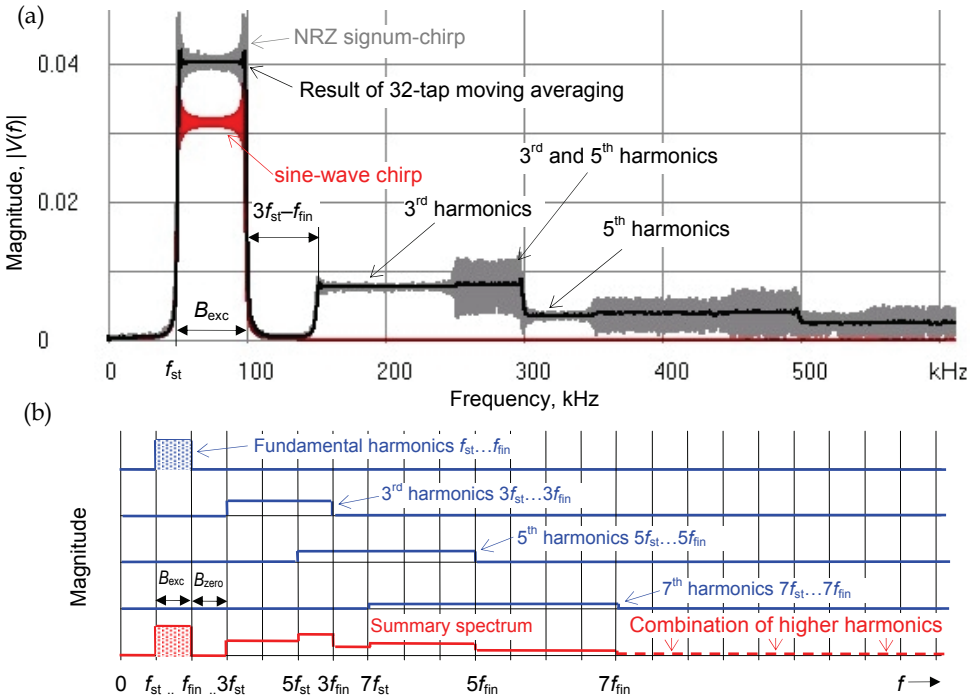


Fig. 13. (a) Amplitude spectrum of signum-chirp with f_{st}=50 kHz and f_{fin}=100 kHz (A=1; L=1000); (b) explanatory diagram of forming the amplitude spectrum at $f_{st}\neq0$

When f_{fin} / f_{st} >> 30 and m >> 1, the value of δ_E = 0.852 and the spectrum has practically uniform energy distribution over the B_{exc}. The relative decrement of every h^{th} step of the averaged amplitude can be calculated from (14) as $\delta_h=(E_{exc}/E_h)^{1/2}=A_1/A_h$. The values for some lower order levels are $\delta_2\approx4.51$, $\delta_3\approx8.44$, and $\delta_4\approx12.47$ – see the approximations given in Fig. 12. For the next levels, a rough approximation $\delta_h\approx4(h-1)+1$ is appropriate for engineering calculations (Min *et al*, 2009).

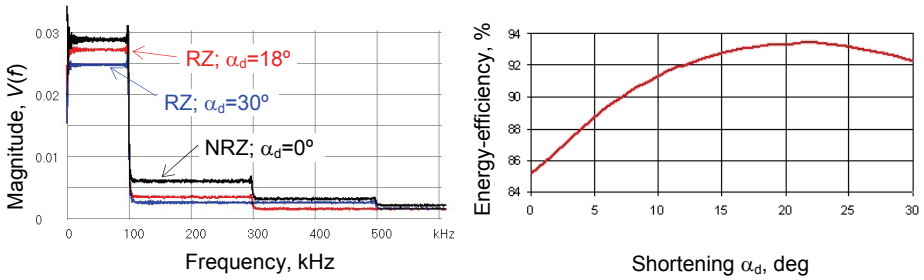
The shape of amplitude spectrum becomes more complicated, if the initial frequency $f_{st} \neq 0$. Though, the spectrum retains the gradual character, the levels do not attenuate monotonically. An example of this kind of spectrum is shown in Fig. 13a, where f_{st}= 50 kHz and f_{fin}=100 kHz. In general, the height and location of spectral levels can be various, depending on the ratio between f_{st} and f_{fin}.

The forming of spectra as the sum of its harmonic components is explained by the draft diagram in Fig. 13b. Fulfillment of the condition $3f_{st} > f_{fin}$ produces the frequency area $B_{zero} = 3f_{st} - f_{fin}$ next to the excitation bandwidth, where the spectrum has almost zero amplitude. In this particular case, the spectral density within the excitation bandwidth is determined only by the fundamental harmonics from f_{st} to f_{fin}. It can be shown, considering the Eqs.(13) –(16), that here the energy-efficiency $\delta_E = 8/\pi^2$.

3.1.3 Ternary chirps

Caused by the absence of particular harmonics (see Sect. 2.2.1), the position and span of amplitude levels of ternary chirps (RZ chirps) differ from the levels of the respective binary chirp. Fig. 14a enables to compare the amplitude spectra of several rectangular chirps with L=1000 at various durations of the zero-level state α_d (shortening).

Let us pay attention to the changed stretch of the amplitude levels. For example, in the case of $\alpha_d = 30^o$, the average amplitude of 2nd level (h=2) stretches uniformly from f_{fin} up to the $5f_{fin}$. As the 3rd harmonic are absent, this 2nd level is formed by the amplitudes of the 5th and higher harmonics up to the frequency $5f_{fin}$.



(a) (b)

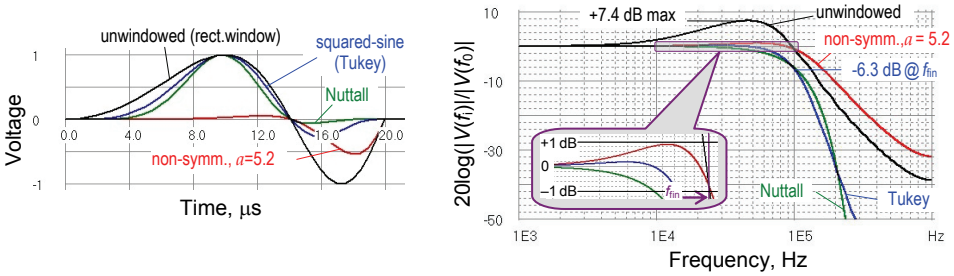
Fig. 14. (a) Amplitude spectra of ternary chirps with f_{fin}=100 kHz (32-tap moving averaging has been applied for smoothing); (b) energy-efficiency of ternary chirps vs. zero-level state (shortening)

Average power of ternary chirps is less than it of respective binary chirps, but the percentage of their useful energy is surprisingly high – over 90% mostly (see also Table 1). Actually, it is possible to generate ternary chirps with any value of the zero-state, but only a few values produce the removing of particular harmonics (see Eq. 6 in Sect. 2.2.1). Nevertheless, we can analyze the energy properties of arbitrary ternary chirp. Employing the equation (12), the maximal energy-efficiency $\delta_E \approx 93.4\%$ was observed at $\alpha_d \approx 22.5^o$, which does not cause disappearance of any odd harmonic component. Dependence of the energy-efficiency on the shortening α_d is plotted in Fig. 14b.

3.2 Windowing of chirps

In the use of some short-time chirps, the problem of flatness of the amplitude spectrum arises (see Fig. 9b). For the single-cycle sine-wave chirp, the maximum overshoot of normalized voltage spectral density is about +7.4 dB inside the chirp bandwidth. To improve the flatness of spectrum, a kind of additional windowing of chirp pulses should be used (in fact, every finite chirp pulse can be dealt as one inside the rectangular window, however we consider this case as the unwindowed one). As a rule, the windowing accompanies with some loss of total energy and power of signals, but still the δ_E can be rather high. On the other hand, frequently the spectral density attains steeper drop-off outside the chirp bandwidth due to windowing. Hence, the optimal choice of windows presumes a certain trade-off always.

In this work, several typical (Hanning, Hamming, Nuttall, etc.) and some specific window functions $F_{win}(t)$ were under study (Barsoukov & Macdonald, 2005). For example, a convenient shaping of amplitude spectra was achieved by implementing window functions $F_{win}(t)=\sin^2(\pi t / T_{ch})$, which can be dealt as a particular case of the Tukey window with the squared-sine lobes and with the tapering time $T_{ch}/2$. Almost perfect shaping was attained using a non-symmetrical windowing in the form of $F_{win}(t)=(t/T_{ch})^a$ with the selectable exponent a (usually a =2…8) (Paavle *et al*, 2010). Some windowing results both in the time and frequency domain are shown in Fig. 15.



(a) (b)

Fig. 15. Effect of windowing to the single-cycle sine-wave chirp, f_{st}=0, f_{fin}=100 kHz, T_{ch}=20 μs: (a) windowed waveforms; (b) normalized spectra of the windowed waveforms

Usually, the deviation about ± 3 dB of the amplitude spectrum inside the B_{exc} is considered as satisfactory for spectral flatness. There are several ways to achieve such the requirement. Using the non-symmetric windowing makes the deviation even less than ± 1 dB accessible (see Fig.15b), but the quantity of useful energy reduces noticeably due to the plain slope of the spectrum at $f > f_{fin}$: for the case, as shown in Fig. 15, the energy-efficiency δ_E=73.1% was observed in simulations. Substantially higher δ_E was achieved by using of Nuttall (δ_E=92.3%) and Hanning (δ_E=95.3%) windows, but the deviation of amplitude spectra inside the B_{exc} is from 0 to -6 dB and from +0.2 to -6 dB, respectively (Paavle *et al*, 2010). The impact of the squared-sine windowing was almost the same as it of the Hanning window.

As a rule, windowing of the generated signal demands some additional power consumption and reduces the total energy-efficiency. However, in the case of very short signals this

drawback can be overcome using the look-up table, which stores the externally calculated values of windowed titlets and which is loaded into the FPGA (Min *et al*, 2011a).

4. Simulation examples

The following examples refer to the linear and quadratic sine-wave chirps and are the results of simulation using the modeling structure in Fig. 7. In all the cases, the chirp excitation from f = 0 to 100 kHz and amplitude A=1 V was implemented. The basic model of EBI was a 5-element impedance $\dot{Z}_1$, consisting of R_0=1 kΩ, R_1=200 Ω, R_2=100 Ω, C_1=30 nF, C_2=20 nF and having the following corner frequencies in the Bode diagram (see Fig. 2b): f_{p1}=2.9, f_{z1}=26.5, f_{p2}=45.2, and f_{z2}=79.6 kHz.

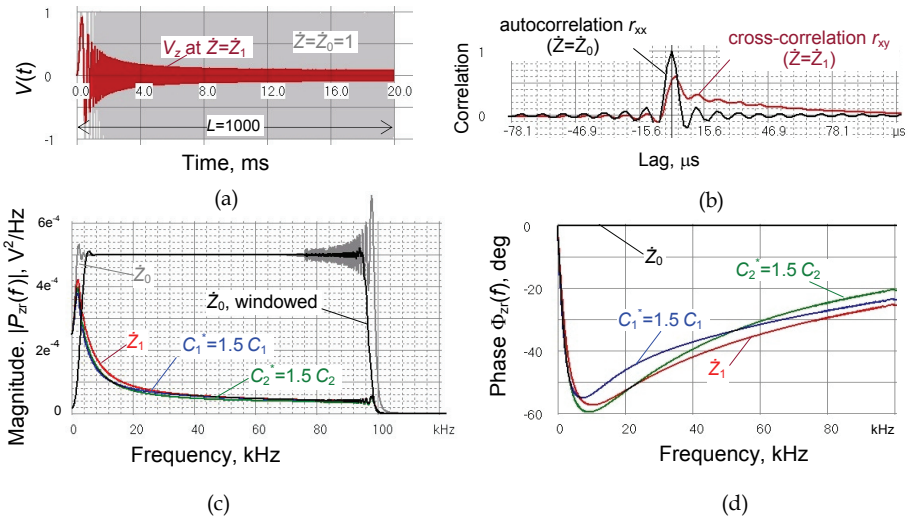


Fig. 16. Examples of the time-frequency analysis of the EBI (sine-wave chirp excitation with L=1000): (a) waveforms of excitation and response; (b) normalized correlation functions; (c, d) cross-power spectral density and phase spectra at variations of the EBI components

Fig. 16 demonstrates simulation results in the case of multi-cycle chirp excitation with pulse duration of T_{ch}=20 ms (1000 cycles). Fig. 16a shows the response voltage $V_z(t)$ on the background of the non-windowed excitation signal. Fig. 16b shows cross-correlation functions in a stretched time scale at the $\dot{Z}$= $\dot{Z}_1$, and at the unity value impedance $\dot{Z}$= $\dot{Z}_0$ =1 (in fact, the latter case corresponds to the autocorrelation function of the excitation signal). Results of the Fourier Transform of the CCF are shown in Figs. 16c and 16d, where one can watch the swing of spectral curves, if the capacitive components of EBI change by +50%. In addition, Fig. 16c demonstrates, how the boxcar-type windowing (here squared-sine Tukey window with the tapering time of $0.05T_{ch}$) reduces the influence of the Gibbs effect.

Intensive fluctuation of spectra, caused by the concurrent higher harmonics, can be more disturbing for signum-chirp excitation (see Figs. 10b, 12, and 13a). This problem can be diminished by proper selection of correlation parameters, e.g., the shorter t_{obs} with rougher frequency resolution smoothes spectrograms essentially (Paavle *et al*, 2008).

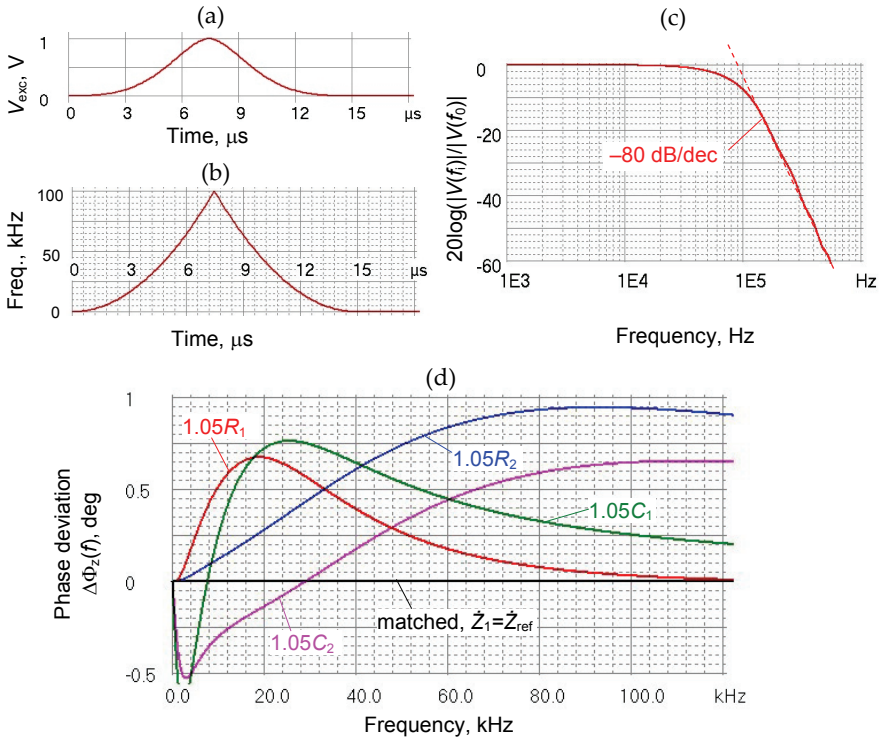


Fig. 17. The use of double quarter-cycle quadratic chirp excitation (L=¼, n=2, T_{ch}=2×7.5 µs, f_{st} =0, f_{fin}=100 kHz): (a) signal waveform; (b) variation of instantaneous frequency; (c) amplitude spectrum of the chirp pulse; (d) phase spectra at matched filtering ($\dot{Z}_{ref} = \dot{Z}_1$) and at changing of the EBI components +5%

For interpreting the variations of frequency responses, the phase spectra should be preferred. First, it is less affected by the spectral fluctuations. Secondly, the phase spectra enable to distinguish changing of object parameters somewhat more clearly than the amplitude ones – compare the respective curves in Figs. 16c and 16d. Nevertheless, substantially better resolution of the EBI for the diagnosing purposes can be achieved by using matched filtering, where instead of $\dot{Z}_{ref}$=1 in the reference channel, a predetermined $\dot{Z}_{ref} \approx \dot{Z}_x$ is used, approximately equal to the impedance $\dot{Z}_x$ under study (see Fig. 6). Joining matched filtering together with analysis of phase spectra allows applying the full scale of some degrees only and permits very good sensitivity for detection of small changes of the object parameters.

Simulation results in implementing of the bidirectional quarter-cycle quadratic chirp excitation for detection of tiny deviations of the EBI vector are presented in Fig. 17. The waveform of excitation pulse and the corresponding frequency run are shown in Figs. 17a and 17b, respectively. Fig. 17c depicts the normalized amplitude spectrum of the excitation signal and Fig. 17d manifests deviation of phase spectra, when the value of single components increases +5% over their initial quality.

Another advantage of the matched filtering is reducing the impact of additive noise, especially the impact of source noise $n_s(t)$. Fig. 18a depicts the noisy input signal, where the generated chirp is affected by the Gaussian noise with the zero mean and variance σ_N^2 – see Fig. 6. The source noise causes some error in the phase spectra at the higher frequencies (Fig. 18b). This error depends on the deviation of component parameters of the equivalent circuit (here C_1 was changed), and on the signal-to-noise ratio (SNR=P_{avg}/σ_N^2), but as a rule, the error remains small even at considerably high level of the source noise.

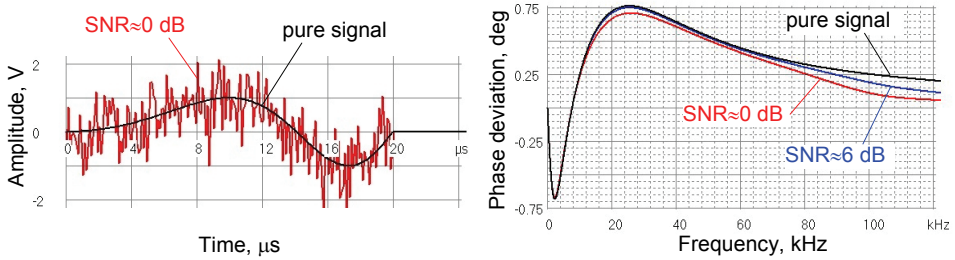


Fig. 18. Linear single-cycle chirp (T_{ch}=20 μs) at the presence of additive source noise: (a) noisy input signal; (b) spectra of phase differences at 1.05C_1 and at different input signal-to-noise ratio (SNR) levels

As the noise from object ($n_z(t)$ in Fig. 6) affects the cross-correlation procedure non-symmetrically, then suppression of noise effect is of importance and serves special attention even at the higher SNR values (Min *et al*, 2011b).

5. Conclusion

The advantages of using chirps as excitation signals in bioimpedance measurement are their wide and almost flat amplitude spectrum together with the independent scalability both in the time and frequency domain – we can choose the frequency range and duration of the excitation pulse almost independently from each other. These features enable to accommodate the generation of excitation signals with the expected properties of the object to be estimated comparatively simply.

It was shown that the shortening of chirp pulses retains the general benefits of chirps – their flat amplitude spectrum within the predetermined frequency range, which permits their implementation in energy-efficient measuring instruments. More than 90% of the generated excitation energy falls into the desired bandwidth even in the case of very short excitation pulses required for providing ultra quick measurement and analysis of dynamic objects. These requirements are obligatory for investigation of objects with rapidly changing parameters (e.g., for identification of fast moving bioparticles such as cells and droplets in high-throughput micro-fluidic systems), and in the devices, in which the low power consumption is important (e.g., wearable units and medical implants). However, shortening of chirp excitation and measurement time should not be excessive. It must be as short as possible to avoid significant impedance changes during the Fourier analysis, but as long as possible to enlarge the excitation energy and to obtain a better signal-to-noise ratio.

The described measurement method, based on the sequentially performed cross-correlation and Fourier Transform, enables the joint time-frequency spectral analysis of bioimpedance. As the time domain cross-correlation function includes the full information about the complex vector $\dot{Z}(\omega)$, only a single FFT-block is necessary for the Fourier analysis and the complete frequency domain determination of the object. Moreover, the cross-correlation procedure assures substantial suppression of noise introduced by the object.

Applying the modification of system architecture with the cross-correlation based matched filtering enables to avoid the impact of noise in much higher degree then a simple cross-correlation. In addition, such the matched filtering permits to increase the sensitivity of measurement, especially through the analysis of phase spectra. Identification and interpretation of relative deviations of the object parameters through the detected tiny phase shifts is of great importance, which in turn can give valuable information about the state and processes in the biological objects.

6. Acknowledgment

This work was supported by the European Union through the European Regional Development Fund, Estonian target-financed project SF0142737s06 and by Enterprise Estonia through the ELIKO Competence Center.

7. References

Barsoukov, E.; Macdonald, J. R. (Eds.) (2005). *Impedance Spectroscopy. Theory, Experiment, and Applications* (2nd ed.). John Wiley & Sons Ltd, Hoboken, New Jersey.

Darowicki, K.; Slepski, P. (2004). Determination of electrode impedance by means of exponential chirp signal. *Electrochemistry Communications*, No. 6, (June, 2004), pp.898-902.

Dwight, H. B. (1961). *Tables of Integrals and Other Mathematical Data*, The McMillan Company, New York, 1961.

Gawad, S.; Sun, T.; Green, N. G.; Morgan, H. (2007). Impedance Spectroscopy Using Maximum Length Sequences: Application to Single Cell Analysis. *Review of Scientific Instruments*, Vol. 78, No.5 (May, 2007).

Grimnes, S.; Martinsen, Ø. G. (2008), Bioimpedance and Bioelectricity Basics (2nd ed.), Elsevier-Academic Press, 2008.

McGhee, J.; Kulesza, W.; Henderson, I. A.; Korczynski, M. J. (2001). *Measurement Data Handling. Theoretical Technique.* Vol. 1, The Technical University of Lodz, Poland.

Min, M.; Parve, T. (2007). Improvement of Lock-in Electrical Bio-Impedance Analyzer for Implantable Medical Devices. *IEEE Trans. Instrumentation and Measurement*, Vol. 56, No. 3 (June 2007), pp.968-974.

Min, M.; Paavle, T.; Annus, P.; Land, R. (2009). Rectangular Wave Excitation in Wideband Bioimpedance Spectroscopy, *Proc. IEEE 4th Int. Workshop on Medical Measurements and Applications (MeMeA2009)*, Cetraro, Italy, May 29-30, 2009, pp.268-271.

Min, M.; Land, R.; Paavle, T.; Annus, P., Parve, T.; Trebbels, D. (2011). Broadband Spectroscopy of Dynamic Impedances with Short Chirp Pulses, *Physiological Measurement*, Vol. 32, No. 7 (July 2011), pp.945-958.

Min, M.; Paavle, T.; Ojarand, J. (2011). Time-Frequency Analysis of Biological Matter Using Short-Time Chirp Excitation. *Proc. of European Conference on Circuit Theory and Design (ECCTD2011)*, Linköping, Sweden, Aug. 29-31, 2011, pp.585-588.

Misaridis, T. X.; Jensen, J. A. (2005). Use of Modulated Excitation Signals in Medical Ultrasound. Part II: Design and Performance for Medical Imaging Applications, *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control*, Vol. 52, No. 2 (Feb. 2005), pp. 192-207.

Müller, S.; Massarani, P. (2001). Transfer-Function Measurement with Sweeps, *J. Audio Eng. Soc.*, Vol. 49, No. 6, (June 2001), pp. 443-471.

Nahvi, M.; Hoyle, B. S. (2009). Electrical Impedance Spectroscopy Sensing for Industrial Processes, *IEEE Sensors Journal*, Vol. 9, No. 12 (Dec. 2009), pp.1808-1816.

Nebuya, S.; Brown, B. H.; Smallwood, R. H.; Milnes, P.; Waterworth, A. R.; Noshiro, M. (1999). Measurement of High Frequency Electrical Transfer Impedances from Biological Tissues. *Electronics Letters Online*, Vol. 35, No. 23 (1999), pp.1985-1987.

Paavle, T.; Annus. P. ; Kuusik, A. ; Land, R. ; Min, M (2007). Bioimpedance Monitoring with Improved Accuracy Using Three-Level Stimulus. *Proc. of European Conference on Circuit Theory and Design (ECCTD2007)*, Seville, Spain, Aug. 26-30, 2006, pp.412-415.

Paavle, T.; Min, M.; Parve, T. (2008). Using of Chirp Excitation for Bioimpedance Estimation: Theoretical Aspects and Modeling. *Proc. of 11th Baltic Electronics Conf. BEC2008*, Tallinn, Estonia, Oct. 6-8, 2008, pp. 325-328.

Paavle, T.; Min, M.; Ojarand, J.; Parve, T. (2010). Short-Time Chirp Excitations for Using in Wideband Characterization of Objects: an Overview. *Proc. of 12th Baltic Electronics Conf. (BEC2010)*, Tallinn, Estonia, Oct. 4-6, 2010, pp.253-256.

Parve, T.; Land, R. (2004). Improvement of Lock-in Signal Processing for Applications in Measurement of Electrical Bioimpedance. *Proc. of Estonian Acad. Sci. Eng.*, Vol. 10, No. 3 (Sept. 2004), pp.185-197.

Pliquett, U.; Gershing, E.; Pliquett, F. (2000). Evaluation of Fast Time-Domain Based Impedance Measurements on Biological Tissue. *Biomed. Technik*, Vol. 45 (Jan.-Feb. 2000), pp.6-13.

Rufer, L.; Mir, S.; Simeu, E.; Domingues, C. (2005). On Chip Pseudorandom MEMS Testing. *Journal of Electronic Testing: Theory and Applications*, Vol. 21, No. 3, pp.233-241.

Sanchez, B.; Vandersteen, G; Bragos, R.; Schoukens, J. (2011). Optimal Multisine Excitation Design for Broadband Electrical Impedance Spectroscopy. *Measurement Science and Technology*, IOP Publishing, 2011, Vol. 22, No. 11, 11 p.

Vaseghi, S. V. (2006)., *Advanced Digital Signal Processing and Noise Reduction*, (3rd ed.), John Wiley & Sons Ltd, Chichester, England, 2006.

11

Simple Signals for System Identification

Paul Annus, Raul Land, Mart Min and Jaan Ojarand
Tallinn University of Technology / Eliko
Estonia

1. Introduction

Event accessible for observation can be investigated by examining timeline of certain measureable values. Usually the process is referred to as time domain analysis of signals. Such a signal can be of many different origins. It could be comprised of values of electrical current or voltage, mechanical displacement or force, value of stock or popularity of politicians, reaction of the patient to medication, and essentially anything which is observable, measurable and quantifiable with reasonable accuracy. Observer will quite intuitively search for certain patterns and periods in signal if phenomena changes along the timeline. Some of them are easily detectable, such as periodic variation of the suns activity; some might appear as random fluctuations at the first glance, such as parameters of the seismic waves in Earth's crust. Of course sources of noise and other disturbances are usually omnipresent and will make matters much more difficult to observe and explain.

It is well known already from early days of modern science that signals can be analyzed in frequency domain as well or instead of time domain analysis. It turns out that for observable events both domains can be easily interchanged or in other words transformed one into other without any loss of information, and even joint time-frequency analysis can be performed.

First known usage of what is essentially known today as fast Fourier transform (FFT) is contributed to Johann Carl Friedrich Gauss (Goldstine, 1977, as cited in Heideman et al., 1984). In 1805 Gauss describes his computationally efficient method for interpolation of the orbits of celestial bodies. Intriguingly it was almost forgotten. Few years later, in 1807, Jean Baptiste Joseph Fourier, while interested in heat propagation, claimed that any continuous periodic signal could be represented as the sum of properly chosen sinusoids during his presentation to the Academy of Sciences in Paris. That claim ignited dispute between the reviewers of his paper: Pierre-Simon Laplace, and Joseph-Louis Lagrange, and delayed the publication until 1822 (Heideman et al., 1984). Joseph-Louis Lagrange's protests were based on the fact that such an approach could not be used to represent signals with sharp corners, or in another words with discontinuous slopes, such as square waves. Dispute lasted almost hundred years until Maxime Bôcher gave a detailed mathematical analysis of the phenomenon and named it after the Josiah Willard Gibbs (Bôcher, 1906). In essence both Lagrange and Fourier were right. While it is not possible to construct signals with sharp corners from sinusoids it is possible to get so close that the difference in energy between these signals is zero. If real signals from nature are concerned instead of exact and purely

mathematical curiosities problem is even smaller. So for all practical signal processing tasks it is indeed possible to state that any real signal can be constructed from sinusoids.

While these transforms, or more precisely their modern counterparts, work very well for signals emanating from natural phenomena, matter can also be investigated by deliberately exciting it with known signals and analyzing its response. This process may be called system identification, synchronous measurement, lock in measurement, or in some application areas something entirely different. The task may even be reversed, in a sense that system can be designed or modified by knowing what output is desirable to certain excitation. It could be filter design in electronics or eigensystem realization algorithm (ERA) in civil engineering, to name a few.

Topic in general is too broad to discuss in one book or even in series of books, and many good papers and books are already written on the topic (Godfrey, 1993; Pintelon & Schoukens, 2001; etc. to name a few), however when low complexity, limited energy consumption and highly optimized measurement systems are targeted new solutions are often warranted, and some of them are briefly discussed in the following pages. So emphasis is given to properties and practical design of custom excitation signals.

2. General considerations

Any system has large set of different parameters, and many subsets of them to be characterized according to requirements imposed by task at hand. Sometimes they can be measured separately and sequentially one at a time, but quite often not. Bulk of system identification theory is based on an assumption that systems are linear and time invariant (LTI), which they are not. Generally certain set of measurements has to be conducted within short enough timeframe for the system to remain reasonable motionless, and with signals which will only very moderately drive system under investigation (SUT) into the non-linear region of operation. Limited magnitudes of excitation signals and the need to consider frequency spectrum of these signals very carefully are both among implications rising from the last requirement. Unfortunately there is also third factor to consider. Disturbances from surrounding world, and noise impact, can effectively render useless any and all of the measurement results, if not dealt with care. Signal to noise ratio (SNR) is often used for numeric quantification of the problem. Similarly known signal to noise and distortion ratio (SINAD) does a little better when covering the problem area and considering likely non-linear behavior of the SUT as well. Shorter excitation time and limited excitation signal energy are always paired with lower SNR.

Therefore it is clear that whenever real systems are characterized, then choice of excitation signals is always a subject of optimization and compromise. Effectiveness of said optimization depends on level of prior knowledge. Things to consider include measurement conditions, SUT itself and the cost of the measurement. Successive approximation and adaptation can be considered when prior knowledge is limited, unless it is one of a kind or very rare event. Fortunately these very rare cases are indeed rare, and furthermore there is usually at least some amount of general prior knowledge available.

Real objects and systems are seldom fully homogenous and isotropic; therefore several measurements from different locations might be warranted for sufficient characterization of the SUT. Possibility of sequential in time measurements from different locations depends

largely on time variance of the parameters of the system. Such a variance could be caused by slow ageing, rapid decay, and fast spatial movement of the SUT relative to the measurement system, as well as by modulation of some of the parameters of the SUT with outside signals. Ideally variance should be excluded by taking readings from different locations simultaneously and rapidly enough. One way to achieve this is to conduct measurements at several slightly differing frequencies. In this case system properties between different points can be separated, and values will not vary much due to almost identical frequencies:

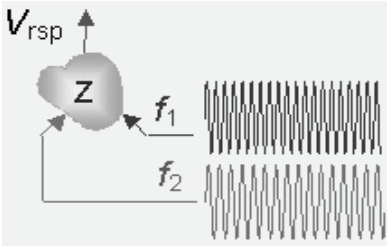


Fig. 1. System identification in case of two simultaneous excitation signal with slightly differing frequencies f_1, and f_2, injected from different points

Similar but different task is accomplished when system properties at vastly different frequencies are of interest. Again sequential in time measurements can be considered, but in case of fast variations in system properties simultaneous multifrequency measurement is essential:

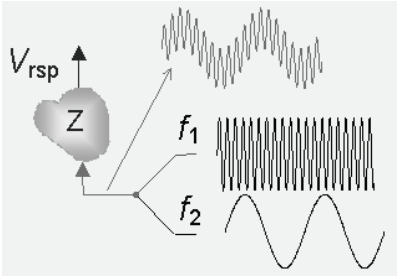


Fig. 2. System identification in case of two excitation waveforms with highly differing frequencies f_1, and f_2, injected from the same point

In even more complex cases multisite and multifrequency measurement are needed simultaneously. Question arises whether it is possible to optimize such a complex measurement by applying different signals and processing methods. Following matter discusses some suitable signals with increasing complexity. For comparison both waveform and frequency content of these signals is given.

Frequency response of any system is comparison of the magnitudes and phases of the output signal spectral components with the input signal components in real world measurement situations. It is best viewed as complex function of frequency. Instead of magnitudes and phases it is often useful to represent the result in Cartesian coordinates. Link is straightforward:

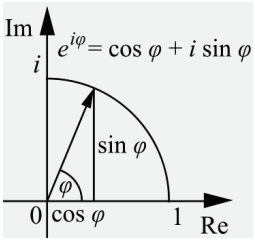


Fig. 3. Complex number with unity magnitude in polar and Cartesian coordinates

If sinusoidal signal with frequency ω, magnitude A and relative phase φ compared to reference signal is multiplied by orthogonal set of reference sinusoids, i.e. $\sin(\omega t)$ and $\cos(\omega t)$, then using simple trigonometric product-to-sum identity it is possible to write:

$$A\sin(\omega t+\varphi)\cdot\sin(\omega t)=\frac{A}{2}\cos(\varphi)-\frac{A}{2}\cos(2\omega t+\varphi) \tag{1}$$

And:

$$A\sin(\omega t+\varphi)\cdot\cos(\omega t)=\frac{A}{2}\sin(\varphi)+\frac{A}{2}\sin(2\omega t+\varphi) \tag{2}$$

From those equations it is clear that if double frequency component and factor of two is disregarded, then first equation can be viewed as giving real part of the complex response, and second equation an imaginary part. When response on only few frequencies is needed, then such a multiplication is often preferred signal processing method both in analog and digital domains. Of course it should be complemented with low pass filtering in order to remove 2ω component. If other than sinusoidal signals are used for excitation and reference, then more frequency components should be considered when calculating response. As long as the reference signal is kept sinusoidal result of the multiplication is still faithful representation of the complex response of the system at the frequency of the reference signal. Some scaling is needed though, due to the fact that magnitude of the fundamental frequency component of the non-sinusoidal signal is different from the magnitude of said signal itself. Unfortunately quite often the reference waveform is far from sinusoidal. Square wave is preferred simply because multiplication of two functions will be replaced by simple signed summing. In analog signal processing domain it will enable to use simple switches instead of sophisticated, error prone, and power hungry multipliers. Same is true in digital domain, where multiply accumulate operation can be replaced with simpler accumulation. Also in this case system response is correct if excitation signal is kept sinusoidal, and in addition to that the system is truly linear. Last part requires very careful analysis, since appearing higher harmonics might coincide with reference signal harmonics, and after multiplication became undistinguishable from the true response. Still, SINAD of the response will degrade even if system can be considered sufficiently linear. If nothing else, then noise will leak into the result, at frequencies which will coincide with higher harmonics of the reference signal. Therefore methods which enable reduction of higher harmonics in reference signal, while keeping it reasonable simple for signal processing, are of utmost importance.

3. Square waves for measurement at single frequency

First the simplest and spectrally worse signal is examined. It is square wave signal or signal which can be described as the sign of $\sin(\omega t)$:

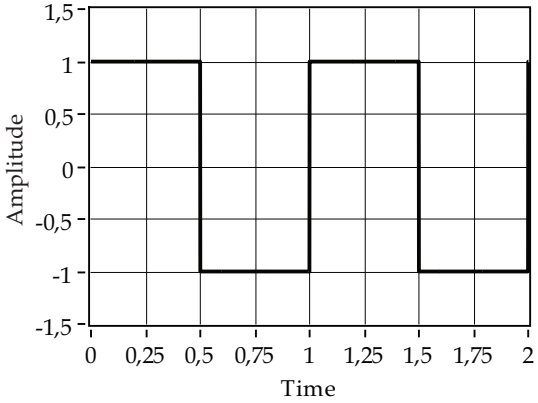


Fig. 4. An odd square wave, with frequency $f = \omega / 2\pi = 1Hz$

Fourier series of this square wave function will contain only sinusoidal members, since function is odd, i.e. $-f(x) = f(-x)$:

$$f(t) = \frac{4}{\pi}\sum_{n=1}^{\infty}\frac{\sin((2n-1)\omega t)}{2n-1} = \frac{4}{\pi}(\sin(\omega t) + \frac{1}{3}\sin(3\omega t) + \frac{1}{5}\sin(5\omega t) + \ldots) \tag{3}$$

It can be viewed as signal with fundamental frequency ω having higher harmonics on odd multiplies of ω. If multiplication of the system response signal with orthogonal references is chosen as signal processing means, and these reference signals are not sinusoids, then all the members of the Fourier series of the response signal are multiplied with all the members of the Fourier series of the reference signals, i.e. not only fundamental components of the signal, but also all higher harmonics. Therefore if those members of the Fourier series coincide in frequency, then the results of multiplication will be added together and become undistinguishable. The measurement is no longer conducted on single well defined frequency, but instead produces results also on all higher harmonics. As it was discussed above it could be largely ignored, if during signal processing multiplication is conducted with sinusoidal signals, unfortunately it is often accomplished with the same rectangular signal instead, and energy form higher harmonics is summed together. Also spectral impact due to non-linearity of the object (or apparatus) cannot be separated from desired response signal anymore. Worst case impact of the coinciding spectral components summed eventually all together can be seen on Fig. 5 with dotted line.

There is another way of looking at how the errors appear (Kuhlberg, Land, Min, & Parve, 2003), by considering phase sensitivity characteristics of the synchronous demodulator (SD). They are easy to draw by varying phase shift φ between two signals. For simple square waves such a characteristic is presented on Fig. 6. For comparison it is drawn together with ideal circle which appears when two sinusoids are multiplied.

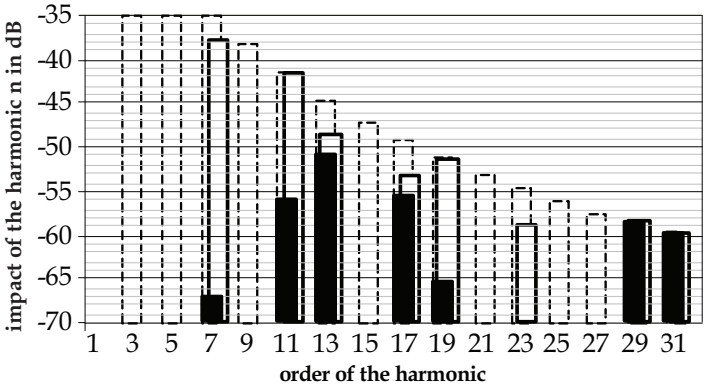


Fig. 5. Worst case relative impact of the higher harmonics to the multiplication result compared to the level of first harmonic in dB. Case of ordinary square wave (dotted line), simple shortened square wave (white boxes), and multilevel shortened square wave (black boxes) (Annus, Min, & Ojarand, 2008)

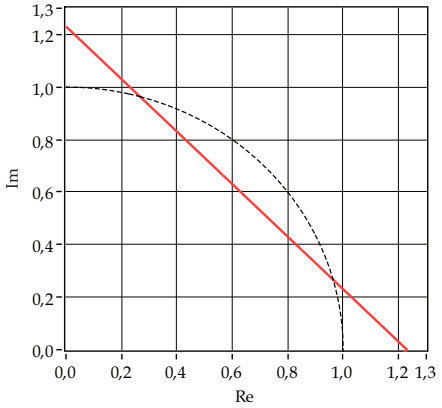


Fig. 6. Quality of synchronous demodulation in case of square wave signals (bold line). For clarity only one quarter is shown

From the Fig. 6 the magnitude error can be easily computed, as the difference between two lines, and Fig. 7. shows this relative magnitude error when square waves are used instead of sinusoids.

Relative magnitude errors of such magnitude as shown in Fig. 7 are generally unacceptable. Not to mention large (several degrees) phase errors in addition to the magnitude error. Only in very specific cases is measurement with pure square waves useful, such as measurement of electrical bioimpedance in implanted pacemakers, where energy constraints are severe.

Fortunately there is very simple method for reducing errors introduced by higher harmonics. Let's consider sum of two square waves with same frequency and amplitude, one of them shifted in phase by β degrees, and another - β degrees. Such a double shift is preferable, since resulting function is again odd. In signal processing odd functions are more

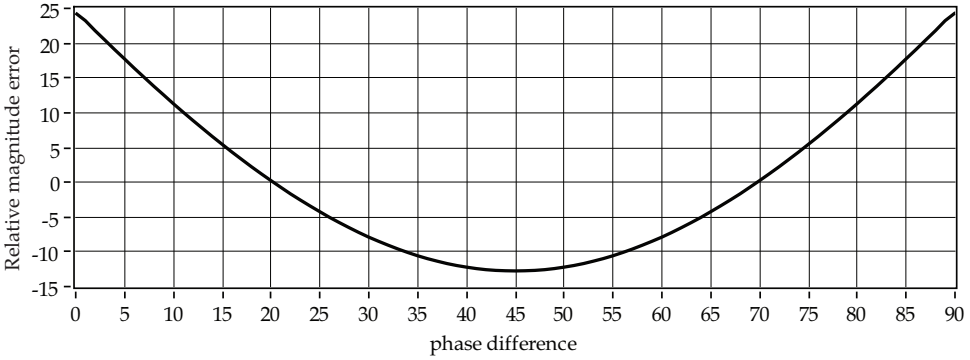


Fig. 7. Relative magnitude error when square wave signals are used during synchronous demodulation

natural, because negative time is usually meaningless, and signals start at *t=0*. Care must be taken that in many mathematical textbooks, and more importantly in different programs, even functions are considered instead. Should the summary phase shift 2β be equal to the half period or odd multiply of half periods of any of the higher harmonic, then such a harmonic will be eliminated from the signal, since sum of two equal sinusoids with 180 degree shift is zero. Such an operation is essentially comb filtering. Main difference of the resulting signal compared with simple square wave is in appearing third level with zero value, so it is reasonable to call them shortened square waves. More generally spectrum of these signals can be derived from Fourier series:

$$f(t)=\frac{4}{\pi}\sum_{n=1}^{\infty}\frac{\cos((2n-1)\beta)\sin((2n-1)\omega t)}{2n-1}=\frac{4}{\pi}(\cos\beta\sin(\omega t)+\frac{\cos 3\beta}{3}\sin(3\omega t)+\ldots) \quad (4)$$

Two of these shortened square waves are of special interest. In order to remove 3rd and 5th harmonics from the signal (as they cause most significant errors) 18 degree and 30 degree shifts are useful. First of them is void of 5th, 15th, etc. harmonics, and second 3rd, 9th, 15th etc. harmonics. It means that impact of these harmonics is drastically reduced. Both of these three level signals with amplitude A are shown on Fig. 8. The third level does not introduce

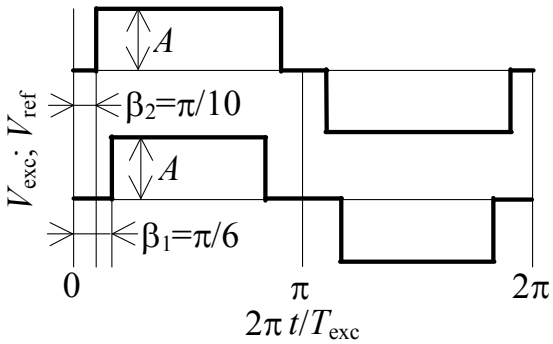


Fig. 8. 18, and 30 degree shortened signals with amplitude A (Min, Kink, Land, & Parve, 2006)

much added complexity from signal generation or processing point of view. Both generation with digital logic, and also synchronous rectification with CMOS switches is straightforward (Min, Kink, Land, & Parve, 2006). If one of them is used as excitation signal and other as rectifying reference, then the result will be much cleaner spectrally when compared to simple square waves, Fig. 5 white rectangles. These two waveforms were chosen, because complete elimination of certain harmonics was desired. Due to that quality of synchronous demodulation is drastically improved, as it can be seen on Fig. 9 and 10. Compared to Fig. 7 improvement is considerable. Not substantial, but still drawback is the need to generate different signals for excitation and synchronous demodulation. What would happen if both signals are shortened by the same amount, and is there an optimum?

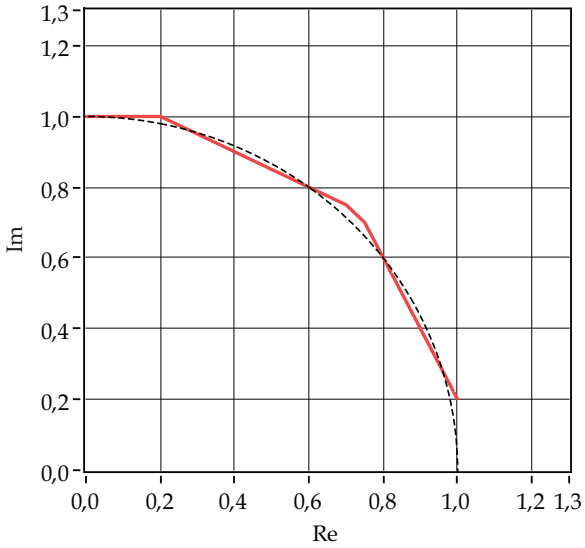


Fig. 9. Quality of synchronous demodulation in case of shortened square wave signals bold line, compared with sinusoidal signals dashed line

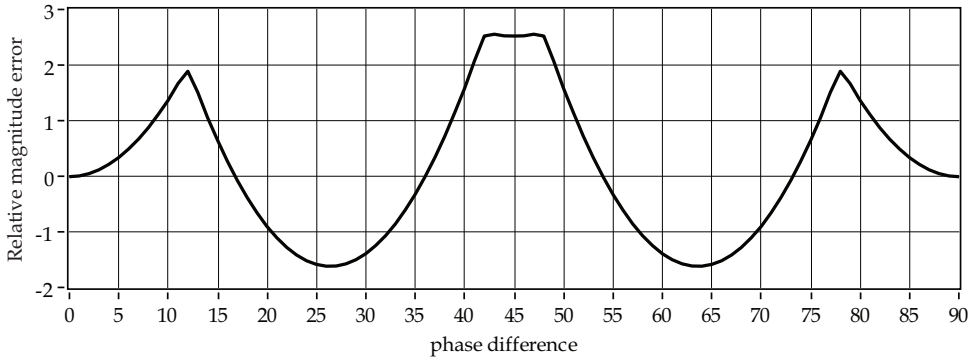


Fig. 10. Relative magnitude error when 18 and 30 degrees shortened square wave signals are used during system identification

If complete elimination of some of the harmonics is not pursued, then arbitrary shift between component square waves is allowed. One possible optimization approach would be to find shortened square wave where minimal energy is leaked into higher harmonics. Relative dependence of the energy of the higher harmonics from shortening angle can be seen on Fig. 11:

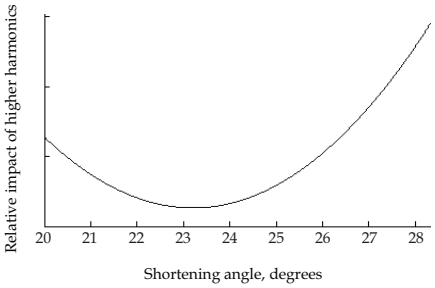


Fig. 11. Relative dependence of the energy of the higher harmonics from shortening angle

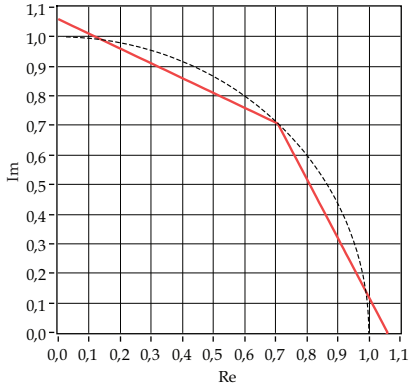


Fig. 12. Quality of synchronous demodulation in case of shortened square wave signals when both are shortened by 22,5 degree bold line, compared with sinusoidal signals dashed line

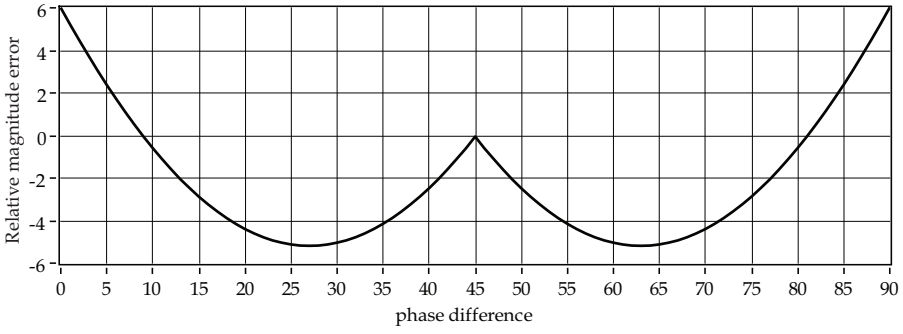


Fig. 13. Relative magnitude error when 22,5 degrees shortened square wave signals are used during system identification

Such a signal could be used for excitation as well as for synchronous demodulator. Since curve is relatively flat around minimum then shortening angels between 22 and 24 degrees produce almost equally good results. Obvious choice for real system identification task would be 22,5 degrees. Compared with previously discussed pair of shortened square waves generation of such a signal requires much lower clock frequency. When 18 and 30 degree shortening is required, then the clock frequency must be at least 30 times higher than the frequency of the resulting signal. With 22,5 degree shortened signal only 8 times higher clock is required. It allows either better energy efficiency by lowering the system clock, or alternatively allows usage of higher frequency signals at the same clock rate. An added benefit is, that noise leakage into result is lower, then in case of 18 and 30 degree shortened pair.

While performing considerably better than simple square waves these shortened signals are far from ideal sinusoid. Could the same summing procedure produce further improvement without much added complexity, if more square waves are added together? The answer is yes. It is enough to add third member into palette consisting already from two square waves shortened by 18 and 30 degrees, namely square wave shortened by 42 degrees. By combining these three signals promising results can be achieved. Three interesting and still simple signals are considered as combinations of previously mentioned summed signals. First and perhaps most obvious is a sum of 18, 30, and 42 degrees shortened square wave signals with signs 1,-1, and 1. Resulting waveform is on Fig. 14, and spectrum of this waveform is on Fig. 15 (Annus, Min, & Ojarand, 2008). It is much cleaner compared to ordinary square wave.

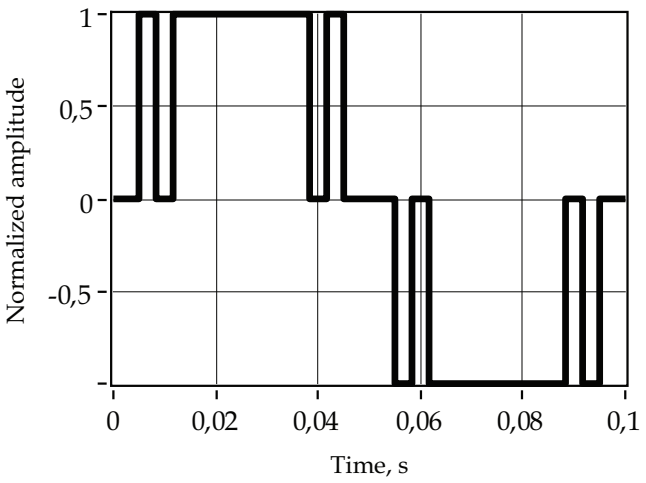


Fig. 14. Resulting waveform from summing of three shortened signals with weights 1, -1, and 1 (Annus, Min, & Ojarand, 2008)

If on the other hand excitation is also shortened square wave, then following pair of signals is suggested (Annus, Min, & Ojarand, 2008). First of them is sum of all three components with coefficients 1,1, and 1, Fig. 16. Spectrum of this summed signal is on Fig. 17.

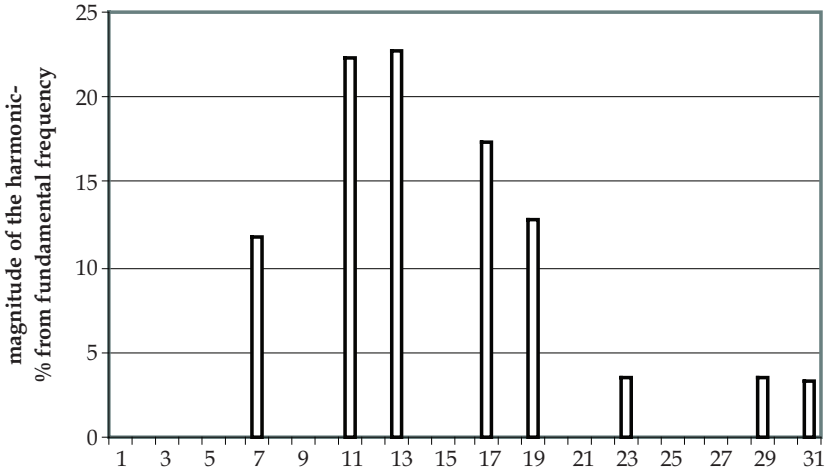


Fig. 15. Spectrum of the signal on Fig. 14 (Annus, Min, & Ojarand, 2008)

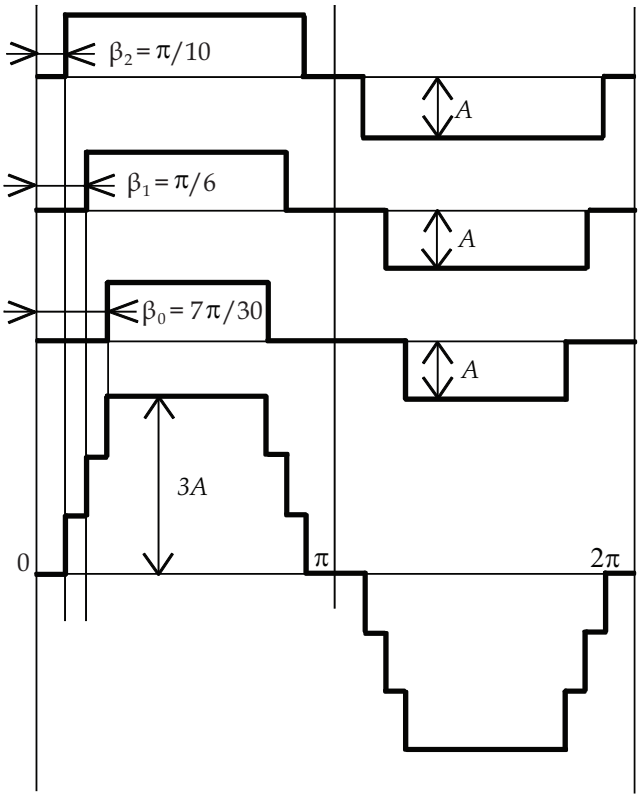


Fig. 16. Sum of three shortened waveforms with coefficients 1, 1, and 1 (Annus, Min, & Ojarand, 2008)

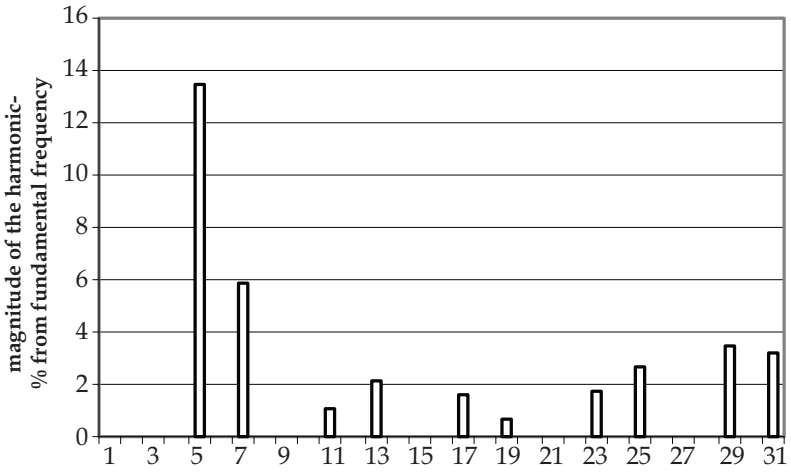


Fig. 17. Spectra of the signal on Fig. 16 (Annus, Min, & Ojarand, 2008)

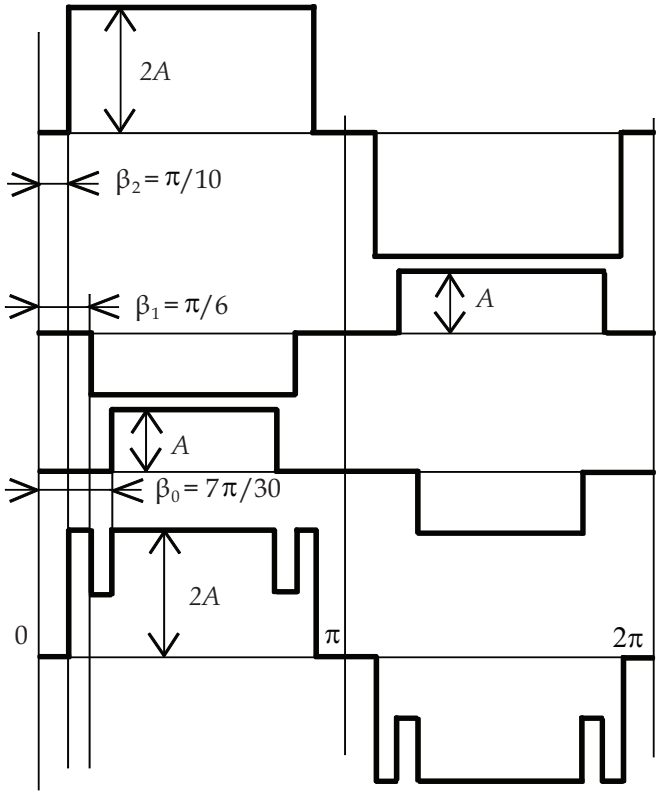


Fig. 18. Sum of three shortened waveforms with coefficients 2, -1, and 1 (Annus, Min, & Ojarand, 2008)

Suitable counterpart summed with coefficients 2, -1, and 1 is on Fig. 18, and spectrum on Fig. 19:

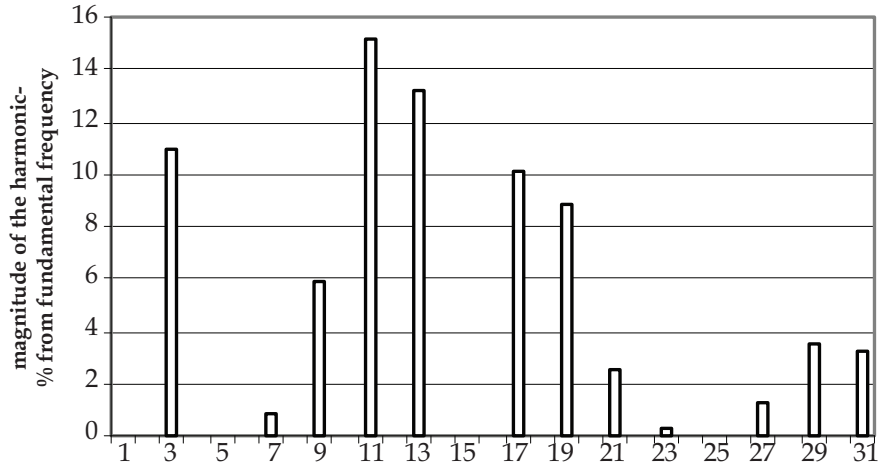


Fig. 19. Spectrum of the signal on Fig. 18 (Annus, Min, & Ojarand, 2008)

Comparison of the worst case multiplication results (Fig. 5) shows significant improvement over previous result. Same improvement can be seen on Fig. 20 as well, where result of synchronous demodulation is shown. Relative magnitude error compared to sinusoid is given on Fig. 21. Nevertheless same clock speed penalty still applies as with simpler solution.

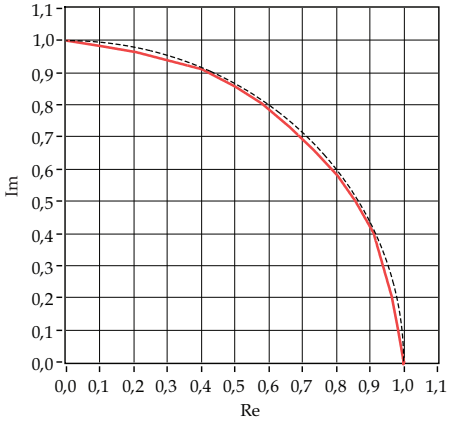


Fig. 20. Quality of synchronous demodulation in case of described multilevel shortened square wave signals

Different way of cleaning square wave spectrally it is described in (Min, Parve, & Ronk, Design Concepts of Instruments for Vector Parameter Identification, 1992). Simple piecewise (over number of system clock periods) constant approximation of the sine wave values is

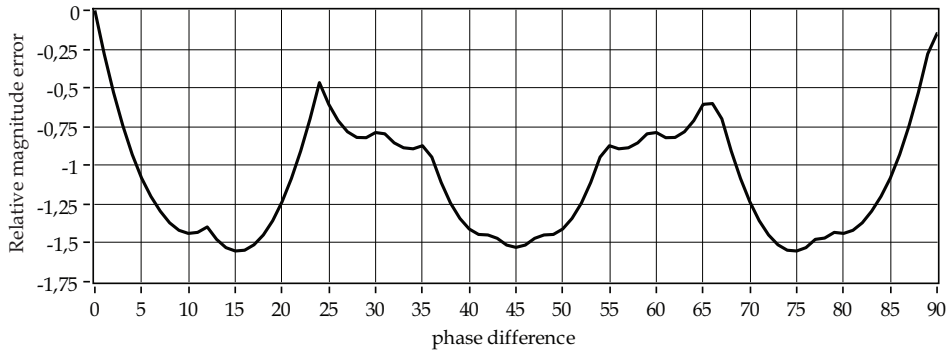


Fig. 21. Dependence of the relative magnitude error from phase difference

used. Waveforms with relatively small number of different levels (3,4,5) are used, and as with already described shortened square wave method different waveforms are suggested for multiplication, resulting in cleaner multiplication product. Values of separate discrete levels are determined according to:

$$a_q = \sin(\frac{\pi}{4m}(2q-1)) \tag{5}$$

Where m is the total number of approximation levels, and q = *0, 1, 2, …, m* is the approximation level number. Spectral composition of these approximated harmonic functions can be found according to the following equation:

$$k_h = 4mi \pm 1 \tag{6}$$

Where k_h is the number of the higher harmonic, which exist in the spectra, and *i=1,2,3,…* If two such signals with number of levels m_1 and m_2 are multiplied, then coinciding harmonics can be found according to:

$$k_c = 4m_1m_2j \pm 1 \tag{7}$$

Where *j=1,2,3,…* If two waveforms with *m=3*, and *m=4* are considered, then first coinciding harmonics are 47th, 49th, 95th, 97th, 143rd, 145th, etc. As with shortened square waves clock frequency should be relatively high, and furthermore these waveforms are relatively sensitive to level errors, which prohibit usage of higher m values, and manifest itself in reappearing higher harmonics.

4. Square waves for multifrequency measurement

If system parameters vary with frequency, as they usually do, single frequency measurement is not enough to fully describe object under test. In complex cases, like measuring electrical properties of biological specimens, sweeping over wide frequency band may be warranted. Sweeping on the other hand is slow, and prohibits examination of faster changes in object under investigation, since transfer function of dynamic system is time dependent. Measurements must be as short as possible to avoid significant changes during

the analysis and, at the same time, as long as possible for enlarging the excitation energy and improving signal to noise ratio.

Chirp signals can be considered to remedy those shortcomings. They will mentioned here only briefly, since more comprehensive overview is in different chapter. Chirp signals, i.e. multi-cycle sine wave based signals in which the frequency increases ('up-chirp') or decreases ('down-chirp') continuously as a function of time, are widely used in radar and sonar applications, acoustic, ultrasonic, optical and seismological studies. The main advantage of chirp signals is their well-defined frequency range and predetermined power spectral density and good crest factor. Rectangular chirps can be used to further simplify signal generation and processing. Moreover, the rectangular waveform has the minimal possible value crest factor (ratio of a peak value to a root-mean-square level) of 1. Two signals are viewed here briefly: binary chirp and ternary chirp (Min et al., 2012). Binary or signum chirp is defined as signum of the sinusoidal counterpart. For linear chirps:

$$sign(ch(t)) = sign(\sin\left[2\pi \frac{B}{T}\frac{t^2}{2}\right]) \tag{8}$$

Where T is duration of the chirp signal, $0 \le t \le T$, and B is bandwidth of the signal. It has crest factor of 1, and it means that energetically the signum chirp is two times more powerful than sinusoidal chirp. Waveform of the binary chirp can be seen on Fig. 22. Third level can be introduced by comparing sinusoidal chirp with two levels instead of one as it is done in case of binary chirp (Fig. 23).

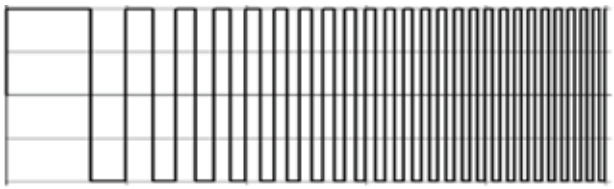


Fig. 22. NRZ or non-return-to-zero binary rectangular chirp pulse (Min et al., 2012)

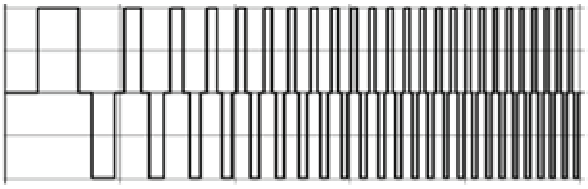


Fig. 23. Return to zero (RZ) ternary rectangular chirp pulse (Min et al., 2012)

Instead of chirp signals other waveforms can be considered as well. A widely used method is to generate a pseudo-random maximum length sequence (MLS). Spectrum of the MLS signal follows a *square(sin(x)/x)* law. Signal processing is usually accomplished by taking circular cross-correlation of the output signal with the excitation MLS. MLS and chirp have however one serious disadvantage – their energy is distributed equally, or almost equally, over the whole frequency band of interest. Therefore, the power spectral density A^2/Hz is comparatively low.

In practice, there is seldom a need to measure at all of the frequencies within the bandwidth simultaneously, except perhaps when system under test is highly resonant. Usually it is enough to know the parameters at several arbitrarily spaced frequencies separately. Therefore, it is reasonable to concentrate the energy of the excitation signals to frequencies of interest instead of using uniform energy distribution over full measurement bandwidth. That can be achieved by summing up several sinusoids:

$$x(t) = \sum_{i=1}^{i=n} A_i \cdot \sin(2\pi f_i + \varphi_i) \tag{9}$$

Unfortunately while single sinusoid is technically feasible signal, and can be reproduced quite accurately, it becomes increasingly costly to use simultaneously many sinusoidal signals. There is another drawback associated with simultaneous use of multiple sinusoidal signals – crest factor. With single sinusoid the crest factor is $\sqrt{2} \approx 1,414$. By summing two or more sinusoidal signals together the crest factor can take many different values, generally bigger then 1,414. Why is crest factor so important? Two reasons are worth considering: nonlinear behavior of the object under investigation, and dynamic range of the measurement apparatus itself. Real objects can rarely be described as linear. It means that different excitation levels do not produce linearly dependent responses. For practical purposes measurement signals are usually kept within narrow range of amplitudes where object behaves approximately linearly. It is clear that such an approximation is better the narrower the range is kept. In worst case high energy pulses can even permanently alter or destroy the object under investigation, and that is certainly not acceptable when performing measurements for example on living human tissue. Also dynamic diapason of the apparatus is limited. From the lower side the limit is set by omnipresent noise signal. If the measurement signal is completely buried in the noise and cannot be restored any more the

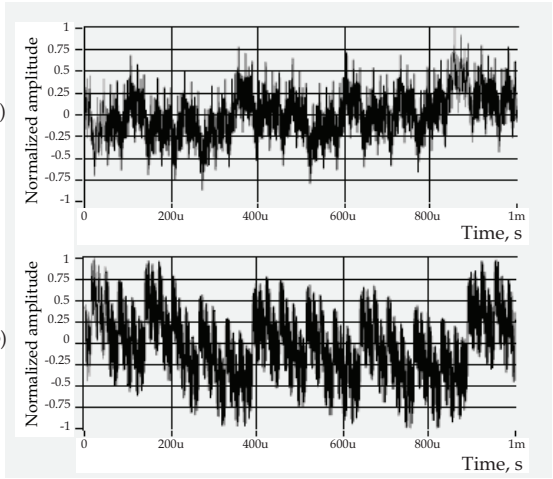


Fig. 24. Multisinusiodal excitation signal (a), containing eleven equal amplitude components, with peak value of 1. For simplicity starting phases of all components are zero. Frequencies in Hz are:1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024. Same signal with some optimization of phases (b); achieved crest factor is 2,5. Please note that the peak values of these two signals are equal

measurement is void. Upper limit is ultimately determined by supply voltage. Form that it is clear that large peaks in excitation signal should be avoided, as well as very low level components which get lost in noise. Fig. 24 is to give an impression of what happens when just eleven sinusoidal signals, one octave apart from each other in frequency, each with equal amplitude, and with random initial phase are summed together. The crest factor of this noise like signal is 3,54, and it is clearly worse than crest factor of a single sinusoid.

Interestingly almost all the parameters of the multisinusoidal signal can be improved considerably by simplifying it drastically (Annus et al., 2011). Such a signal is derived from the original multisinusoidal signal by detecting its zero crossings:

$$sign(x(t)) = sign\sum_{i=1}^{i=n} A_i \cdot \sin(2\pi f_i + \varphi_i) \tag{10}$$

In case of n= *11* the resulting waveform is shown on Fig. 25.

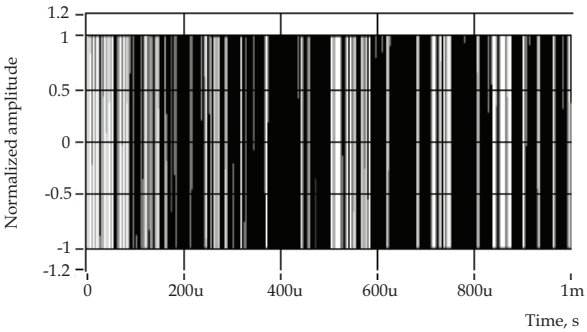


Fig. 25. Typical binary multifrequency signal with eleven components and random phases

Clearly multifrequency binary signal is easier to generate then multisinusoidal signal, and has far superior crest factor of 1. Minor drawback can be seen on Fig. 26, where spectrum of such a signal is shown. So called "snow" lines appear between the wanted frequency components, and roughly 30% of the total energy is lost for measurement:

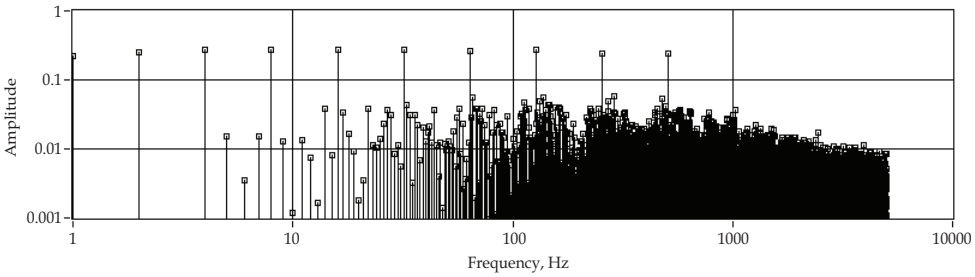


Fig. 26. Spectrum of binary multifrequency signal with ten components and random phases

Truth is revealed when spectrum of optimized multisine is drawn together with spectrum of the binary counterpart:

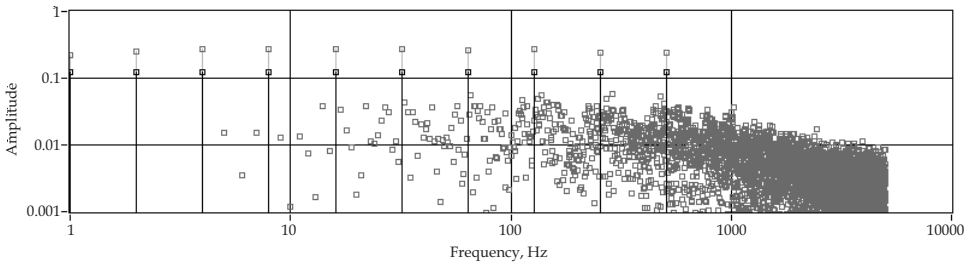


Fig. 27. Magnitudes of spectral lines in binary multifrequency signal (grey squares) together with spectrum of multisinusoidal signal (black squares)

Useful spectral components in binary multifrequency signal contain 1,34927 times more energy than in case of optimized multisine, or 4,32933 times more than the multisine above without optimization. At first glance there is another drawback, since magnitudes of useful components are not exactly equal anymore. Fortunately said magnitudes can be easily equalized by iteratively manipulating magnitudes of the components of the original multisinusoidal signal. Residual error is generally well below one percent. Furthermore almost arbitrary magnitudes of spectral components can be achieved (Fig. 28).

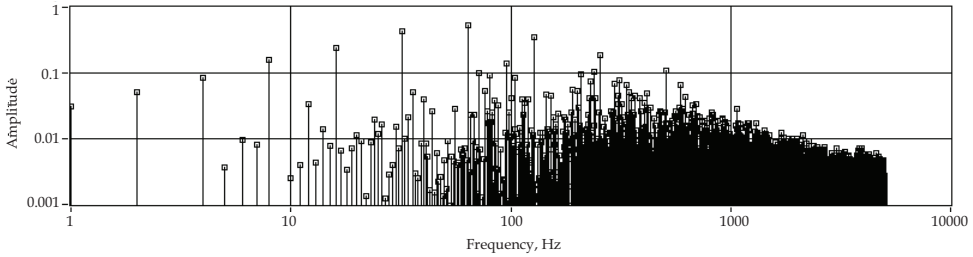


Fig. 28. Magnitude control example

When two level comparison is introduced instead, then energy leakage into unwanted components can be reduced. Such a ternary multifrequency signal can be seen on Fig. 29.

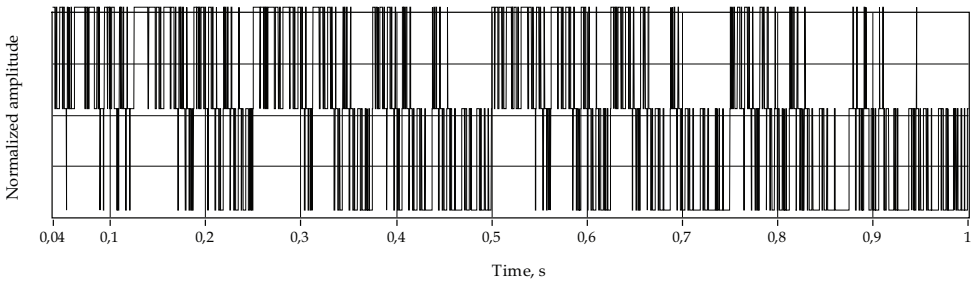


Fig. 29. Ternary multifrequency signal. Comparison levels are set on +/- 0,23V

Comparison of the spectrum of the original multisinusoid with ternary signal from Fig. 29 can be seen on Fig. 30.

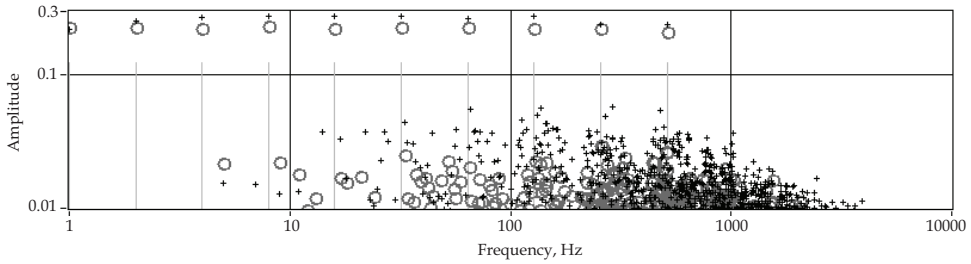


Fig. 30. Magnitude spectrum of the ternary multifrequency signal (grey rings) compared with magnitudes of multisinusoid components (grey lines), and binary multifrquency signal (crosses)

5. Conclusion

With some minor drawbacks it is possible to construct relatively simple square wave signals in order to replace more sophisticated sinusoidal or arbitrary waveforms when system identification is warrantied. Simplest square wave nevertheless might not be sufficiently good measurement signal. By adding few more levels situation can be improved considerably. These signals can be used to replace single sinusoids, chirps, and arbitrary sums of sinusoids. Generally it is a good idea to choose best signal for given identification task. Some choices are shown, and reasoning behind theme given.

6. Acknowledgment

This research was supported by the European Union through the European Regional Development Fund and through the projects of Eliko Competence Centre, and Centre for Research Excellence CEBE.

7. References

Heideman, M.; Johnson, D.; Burrus, C. (1984). Gauss and the history of the fast fourier transform, ASSP Magazine, IEEE , vol.1, no.4, pp.14-21

Maxime Bocher (1906). Introduction to the Theory of Fourier's Series. The Annals of Mathematics, Second Series, Vol. 7, No. 3, pp. 81-152

Keith Godfrey (1993). Perturbation Signals for System Identification (Prentice Hall International Series in Acoustics, Speech, and Signal Processing)

Rik Pintelon, Johan Schoukens (2001). System Identification: A Frequency Domain Approach. Wiley-IEEE Press; 1 edition

Annus, Paul; Min, Mart; Ojarand, Jaan (2008). Shortened square wave waveforms in synchronous signal processing. IEEE International Instrumentation and Measurement Technology Conference Proceedings: 2008 IEEE International Instrumentation and Measurement Technology Conference (I2MTC 2008), 12-15 May 2008, Victoria, British Columbia, Canada, 2008. , 1259 - 1262.

Min, M.; Parve, T.; & Ronk, A. (1992). Design Concepts of Instruments for Vector Parameter Identification. IEEE Transactions on Instrumentation and Measurement, Volume: 41 Issue:1, pp. 50 – 53

Min, Mart; Parve, Toomas; Land, Raul; Annus, Paul (2012). Scalable Impedance Spectroscopy: comparative study of sinusoidal and rectangular chirp excitations. In: Lecture Notes on Impedance Spectroscopy v. 2 Measurement, Modeling and Applications – Olfa Kanoun (ed), Taylor & Francis Group, London, 33-38.

Annus, P.; Min, M.; Ojarand, J.; Paavle, T.; Land, R.; Ellervee, P.; Parve, T. (2011). Multisine and Binary Multifrequency Waveforms in Impedance Spectrum Measurement – A Comparative Study. 5th European IFMBE MBEC 2011, Budapest, 14-18 Sept., 2011, 1265 - 1268.

Part 3

Image Processing

12

Invariant Nonlinear Correlations via Fourier Transform

Ángel Coronel-Beltrán[1] and Josué Álvarez-Borrego[2]
[1]*Universidad de Sonora, Departamento de Investigación en Física,*
[2]*CICESE, División de Física Aplicada, Departamento de Óptica,*
México

1. Introduction

A great advance has been done in optical and digital pattern recognition since the first introduction of the well known classical matched filter (CMF) (Vander Lugt, 1964), which was the cornerstone for the developments of new and more effective filters. However, the use of the CMF is inefficient because the output correlation peak is degraded drastically by the geometrical distortions of the target, such as scale and rotation changes. Many attempts have been done to achieve distortion-invariant sensitivity pattern recognition. To overcome these problems, the Fourier-Mellin transform was introduced (Casasent, 1976a, 1976b, 1976c) for scale and rotation invariance. The realization of the Mellin transform is given by a logarithmic polar mapping of the input image followed by a Fourier transform.

Some studies have been done confirming the invariance to position, scale and rotation in the visual cortex in primates and humans (Schwartz, 1977, 1980, 1984) through a logarithmic polar mapping from the retina to the visual cortex as is known in the optical and digital pattern recognition. In this last aspect, for the majority of the digital systems the process is very different to the sensorial visual system, because to its natural complexity composition. The digital images are obtained in different environment conditions in an optical-digital sensor that causes several problems for the identification and characterization of the object using computerized vision systems, also the study is limited to static objects, and only bi-dimensional images are considered.

The scale transform has been proposed (Cohen, 1993, 1995) and is very suitable to scale sensitivities instead of the well known Mellin-Fourier transform. It has been applied to different areas such as image filtering and denoising (Cristóbal et al., 1998), in the identification and registration of some alphabetic letters and diatoms by computing the power cepstrum of the log-polar scale mapping (Pech-Pacheco et al., 2000) and in the automatic identification of phytoplanktonic algae (Pech-Pacheco et al., 2001) and in the analysis of letters scaled and rotated (Pech-Pacheco et al., 2003).

In this work, we obtain some results with a new computational algorithm for the recognition of several objects, independently of its size, angular orientation, displacement and noise. We use the scale transform and the k-th law nonlinear filter (Vijaya Kumar & Hassebrook, 1990) with a nonlinearity strength factor of k=0.3 (Coronel-Beltrán & Álvarez-Borrego, 2008). A kth

law nonlinear filter is introduced to realize the digital invariant correlation that gives us information on the similarity between different objects. The nonlinear strength factor is the exponent in the modulus of the Fourier transform of the object to be recognized in the expresion of the nonlinear filter. This kind of filter has advantages compared with the phase only filter (POF) (Horner & Gianino, 1984) and other linear filters, due to its great capacity to discriminate objects, the maximum value of the correlation peak is well localized, narrow, with small sidelobes and the output plane is less noisy. In other works, a *k*th law joint transform correlator is used (Javidi, 1989a, 1989b, 1990), instead of a *k*th law nonlinear filter, in an optical coherent system (Javidi, 1989c), with better results, where the nonlinear characteristics are used to generate the nonlinear filter with an electrically addressed Spatial Light Modulator (SLM) located in the Fourier plane of the optical correlator. It is shown that both types of nonlinear filter produce better correlation performance compared with linear filters (Javidi, 1990). In our case, we used the *k*th law nonlinear filter because we are treating an invariant digital system and not an optical system. However, in an optical system it is difficult to implement the scale transform in order to have the invariances to scale and rotation.

In this work, the nonlinearity affects the Fourier magnitude of both the input signal and the reference signal. We applied this method for the recognition of the alphabetic letters in Arial font type and plain style with one rotational degree and one percent increments for rotation and scaling, respectively, until a complete rotation of 360 deg and ±25% scale variation is achieved. When the input scene is in presence of additive Gaussian and impulse salt-and-pepper (S&PP) noise we analyzed the performance of the nonlinear filter using the discrimination capability (DC) metric. The results were improved significantly when we introduced in the correlation digital system the nonlinear Spearman correlation.

The material of this work is organized as follows: in section 2, the preliminary concepts used to construct the invariant digital system are introduced. In section 3, the performance metrics used to evaluate the filters in the correlation output plane are presented. In section 4, the nonlinear invariant correlation method is described. Section 5 shows the results of some examples using the digital invariant correlation system with a nonlinear filter. Moreover, the results obtained by this new methodology were compared using the nonlinear and phase only filters. In section 6, a target immersed in Gaussian and S&PP noise is treated and the discrimination coefficients are shown. In section 7, the digital invariant system with a nonlinear filter in conjunction with a nonparametric statistical method is presented. In section 8, a comparative analysis of different font types in plain and italic style letters and the effects of five foreground/background color combinations using an invariant digital correlation system with a nonlinear filter is given, and in section 9 the conclusions of our digital system for nonlinear invariant correlation are given.

2. Preliminary concepts

In this section, we provide a brief background of the principal concepts used in this work. These are: the nonlinear correlation filter, the phase only filter and the scale transform. In the first, the *k-th* law is implicit in this kind of filter. In the second, we show that this filter is a consecuence of the most general filter defined as the nonlinear filter with k=0. And for the last, we have the scale transform as a special case of the Mellin transform that form a family of Mellin transforms depending of a real fixed parameter.

2.1 The *k-th* law nonlinear filter

A nonlinear correlation filter denoted by H_{NLF} is expressed as (Vijaya Kumar & Hassebrook, 1990)

$$H_{NLF}(u,v) = |S(u,v)|^k \exp[-j\phi(u,v)], \quad 0 < k < 1 \tag{1}$$

where $|S(u,v)|$ is the modulus of the Fourier transform of the object to be recognized, $j = \sqrt{-1}$, k is the nonlinearity strength factor and $\phi(u,v)$ is the phase of the filter´s Fourier transform. We can manipulate the discrimination capabilities of the nonlinear correlator system by varying the k value between 0 and 1 to determine the best k value for the nonlinear filter.

2.2 The phase only filter

From equation (1), taking k=0, we have the phase only filter that is denoted as (Horner & Gianino, 1984)

$$POF = \exp[-j\phi(u,v)]. \tag{2}$$

2.3 The scale transform

The scale transform is a special case of the Mellin transform, where the last is defined as (De Sena & Rocchesso, 2007)

$$M_f(p) = \int_0^\infty f(x)\, x^{p-1} dx, \tag{3}$$

in the complex variable $p = -jc + \beta$, with the fixed parameter $\beta \in \mathbb{R}$ and the independent variable $c \in \mathbb{R}$. This family of transforms is called the β-Mellin transform. The real part of the complex variable p is parameterized, for the case $\beta = 1/2$ we have the scale transform. Other values for β are possible, for $\beta = 0$ we have the compression/expansion invariant transform, and for $\beta = -1$ we have the form invariant transform. Therefore, the scale transform is a restriction of the Mellin transform on the vertical line $p = -jc + 1/2$.

We introduce the scale transform due to its property of invariance to size changes. The scale transform and its inverse are given in one-dimension by (De Sena & Rocchesso, 2004)

$$D_f(c) = (2\pi)^{-1/2} \int_0^\infty f(x) \exp[(-jc - 1/2)\ln x]\, dx, \tag{4}$$

and

$$f(x) = (2\pi)^{-1/2} \int_{-\infty}^\infty D_f(c) \exp[(jc - 1/2)\ln x]\, dc. \tag{5}$$

We used the 2-D scale transform in polar coordinates (r,θ) with the log of the radial coordinate $\lambda = \ln r$, that is expressed as (Cristóbal et al., 1998)

$$D(c_\lambda, c_\theta) = (2\pi)^{-1/2} \int_0^\infty \int_0^{2\pi} \exp(\lambda/2) f(\lambda,\theta) \exp[-j(\lambda c_\lambda + \theta c_\theta)]\, d\lambda d\theta, \tag{6}$$

where the non-separable scale transform implementation has been utilized in order to maintain the invariance to rotation (Pech-Pacheco et al., 2003).

3. Metrics used in performance evaluation

In this section, we present two well-known metrics used to evaluate the performance of the filters in the correlation output plane. These are: the peak-to-correlation energy and the discrimination capability metrics. The first is used for objets free of noise and the second for objetcs immersed in noise.

3.1 The peak-to-correlation energy (PCE)

The peak-to-correlation energy PCE performance metric is defined as (Javidi & Horner, 1994)

$$PCE = \frac{|E\{c(0,0)\}|^2}{E\{|c(x,y)|^2\}}, \tag{7}$$

where the numerator is the correlation peak intensity expected value and the denominator is the mean energy expected value in the correlation plane.

3.2 The discrimination capability (DC)

If we consider that an object is embedded in a noise background, the discrimination coefficient that is a modified version of the discrimination ratio (Vijaya & Hassebrook, 1990) is given by

$$DC = 1 - \frac{|C^N(0,0)|^2}{|C^{OBJ}(0,0)|^2}, \tag{8}$$

where the correlation peak produced by the object is C^{OBJ}, and the highest peak of just the noise background is C^N.

4. The nonlinear invariant correlation method

In this section we present the methodology used in this work of the digital invariant correlation system with a nonlinear filter. First, we describe in each step the process to obtain the nonlinear filter of the target. Secondly, the invariant correlation system with the nonlinear filter is described. In both cases, the algorithms are represented with simplified block diagrams.

4.1 Obtaining the digital invariant system with a nonlinear filter (DISNF)

The steps to obtain the DISNF are showed in Fig. 1. In step (1), we have the original image, $f(x,y)$ consisting of the target. The fast Fourier transform (FFT) is calculated in step (2). The modulus of the Fourier transform denoted as $|F(u,v)|$, is obtained in step (3); in this way the displacement of the input image is not affected in the Fourier plane according to the well known shift theorem (Goodman, 2005). Next, we applied a parabolic filter (step 4) (Pech-Pacheco et al., 2003). This type of filter attenuates low frequencies and passes high frequencies that increases the sharpness in the details of the object.

The next step is to introduce the scale factor given by $r^{1/2}$ (step 5) (r is the radial spatial frequency, the origin of which lies at zero frequency in the optical representation of the

Fourier spectrum). This process is what differentiates the scale transform from the Mellin transform.

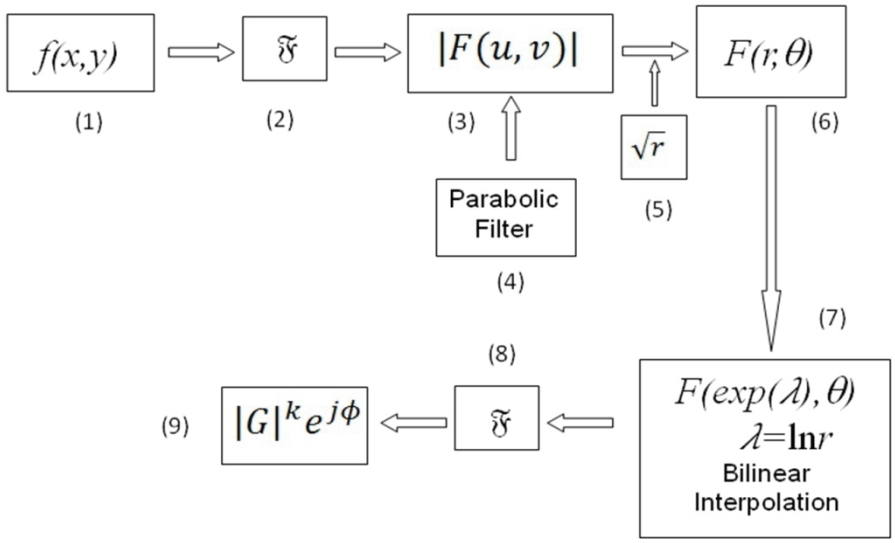


Fig. 1. Simplified block diagram for obtaining the nonlinear filter

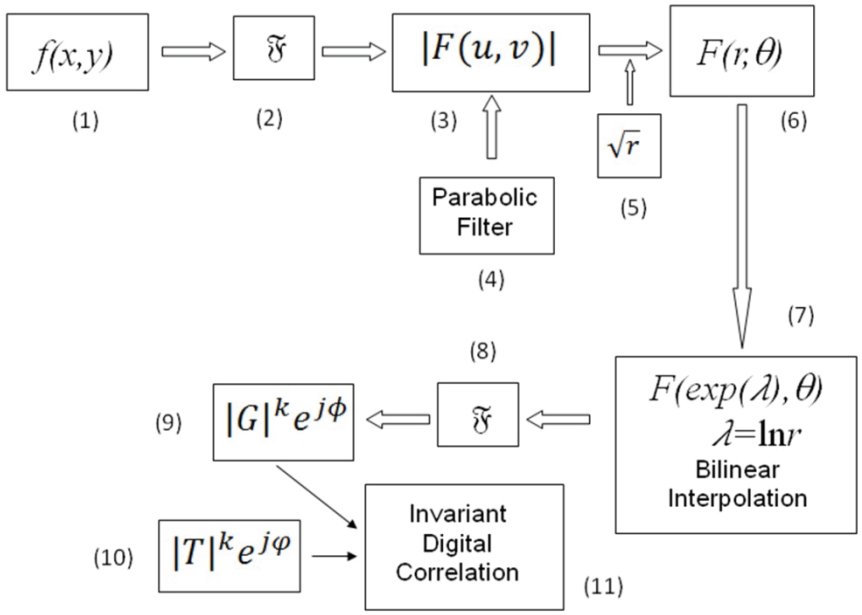


Fig. 2. Simplified block diagram representing the invariant correlation system with a nonlinear filter

Cartesian coordinates are mapped to polar coordinates to obtain the rotation invariance (step 6). In this step we introduced a bilinear interpolation of the first data of coordinate conversion. This is done to avoid the aliasing due to the log-polar sampling. A logarithmic scaling is made in the radial part in polar coordinates (step 7). And taking the FFT (step 8), we obtain a filter with invariance to position, rotation and scale (step 9). In this step we write the filter in a similar manner to equation (1), where $|T|$ and φ are the modulus and the phase of the Fourier transform of the object to be recognized, respectively, after to consider the invariance.

4.2 The nonlinear invariant correlation

The steps to realize the invariant correlation with a nonlinear filter are shown in Fig. 2. The problem image which contains or not the target is the input image (step 1). From the step (1) to (9), the procedure is the same as in Fig. 1. The step (9) shows the nonlinear information of the problem image, where $|G|$ and ϕ are the modulus and the phase of the Fourier transform of the problem image, respectively, after to consider the invariance. The steps (9) and (10) show the correlation procedure to obtain the invariant digital correlation (step 11) to position, rotation and scale using a nonlinear filter. The linear and nonlinear filter inside this invariant process is a combination more powerful to recognize different patterns with different scale and rotations.

5. Computer simulations

In this section, we present a numerical statistical experiment to find the best k nonlinearity strength factor value for the nonlinear filter.The results were box plotted, peak-to-correlation energy (PCE) *vs* k, one for the rotation and the other for the scale. Image **A** was used as the target (Fig. 3). Obtaining that the results for the maximum k value were the same as in (Coronel-Beltrán & Álvarez-Borrego, 2008), that is k=0.3.

Fig. 3. Images used for obtaining the best k value. Image **A** was used like the target

A box plot graph for the peak-to-correlation energy PCE *vs* the nonlinearities *k* values is shown in Fig. 4 for the rotation case, from *k*=0 to *k*=1 for the target **A** correlated with the 15 problem images denoted for **A**, **B**, **C**,…**O**, shown in Fig. 3, rotated, with one rotation degree increment, from 0 to 179 rotation degrees. This graph shows the mean value with one standard error (±SE) and two standard errors (±2*SE) for the PCE. The number of images processed for each *k* value was 2700. Because we select 11 different *k* values, 0, 0.1, 0.2, …, 1.0, so we have a total of 29700 statistically processed images.

From Fig. 4, we observe that for values $0.1 \leq k \leq 0.4$ we have the best performance of the nonlinear filter.

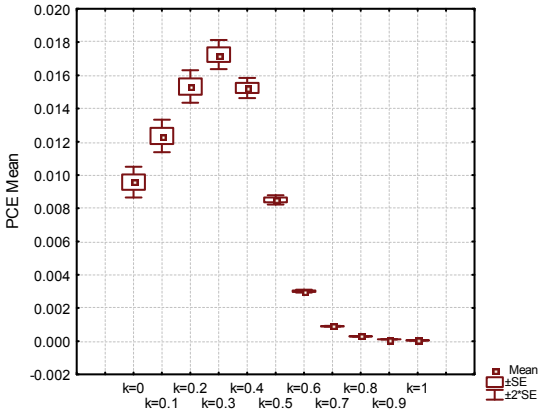


Fig. 4. Box plot graph for the peak-to-correlation energy PCE vs the nonlinearities k values, for the image **A** correlated with the 15 rotated problem images in Fig. 3

In a similar manner, a box plot graph for the peak-to-correlation energy PCE *vs* the nonlinearities *k* values is shown in Fig. 5 for the scale case, from *k*=0 to *k*=1 for the target **A**

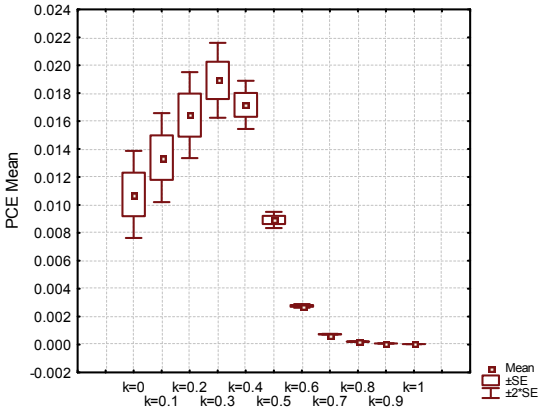


Fig. 5. Box Plot graph for the peak-to-correlation energy PCE *vs* the nonlinearities k values, for the image **A** correlated with the 15 scaled problem images in Fig. 3

correlated with the 15 problem images scaled, with one percent increment, from 75% to 125%. This graph shows the mean value with one standard error (±1*SE) and two standard errors (±2*SE) for the PCE. The number of images processed for each k value was 765. We have the same 11 different values of k used for the rotation problem, hence we have a total of 8415 statistically processed images.

From Fig. 5, we observe that for values $0.1 \leq k \leq 0.4$ we have the best performance of the nonlinear filter again.

For the rotated and scaled cases, the result for the best k value was the same, k=0.3.

5.1 DISNF applied to alphabetic letters

Some results obtained using a DISNF for testing this method for changes of rotation and scale of the letter **E** are presented. In other example the cross-correlations were calculated for a nonlinear and a phase only filters and its performance was compared.

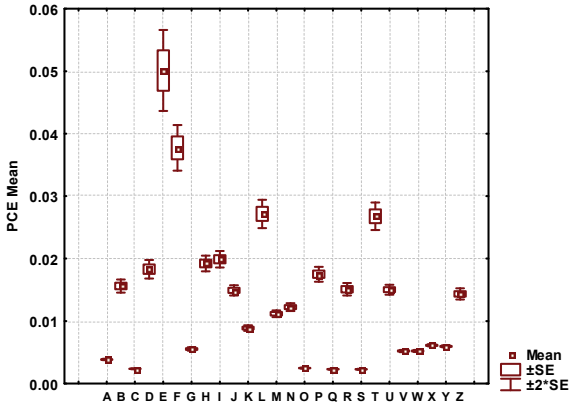


Fig. 6. Performance of the filter **E** for rotation

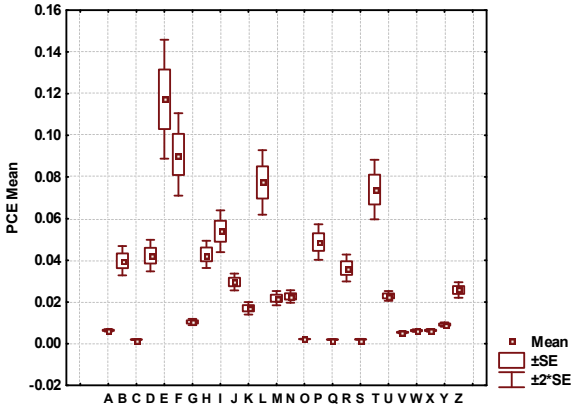


Fig. 7. Performance of the filter **E** for scale from 75% to 125%

We used the *k* strength factor value for the nonlinear filter using the letter **E** for rotation and scale, *k*=0.3. The results of the filter **E** for rotation and scale are shown in Figs. 6 and 7. The filter is correlated with each one of the 26 alphabet letter, with one rotation degree increment, from 0 to 359 deg and scaled from 75% to 125% with one percent increment considering that the filter is 100% in scale.

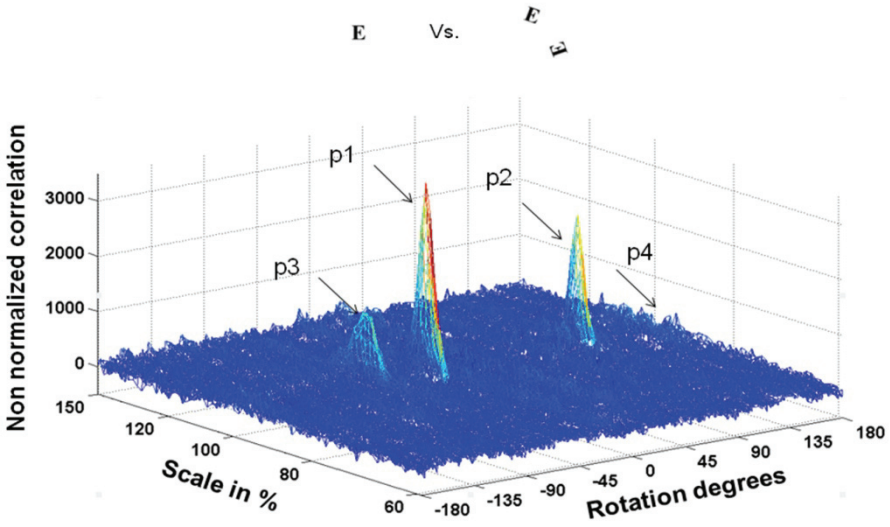
With respect to rotation the algorithm has a good performance to ±2*SE when we want to recognize the **E** letter (Fig. 6). More results were obtained using different characters like target (Coronel-Beltrán & Álvarez-Borrego, 2008). From Fig. 7 we observe an overlap of the letter **E** with the **F** and **L** letters, but, to ±1*SE the invariant correlation had a good performance.

5.2 Comparison of an invariant digital system using nonlinear and phase only filters

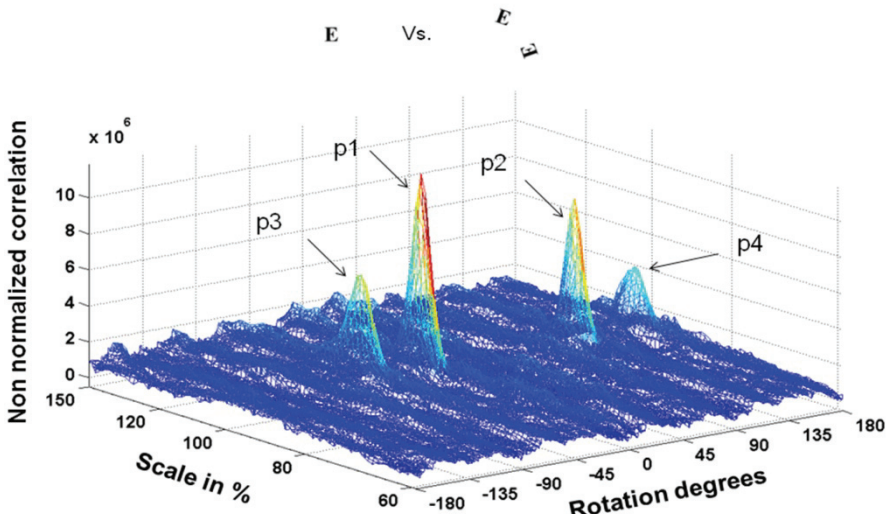
We present another correlation examples using a nonlinear filter with the same *k*=0.3, and compared with the phase only filter. Figs. 8 and 9 show the graphs of the output correlation performance of these filters where the filter and the problem image are the letters E. We observe a less noisy correlation plane (Fig. 8a) with peaks well defined p1, p2, and p3 when a nonlinear filter is used. The PCE value was 0.0215. An output plane correlation with more background noise is presented when in the invariant correlation a phase only filter is used (Fig. 8b). For this case the PCE value was 0.0057.

Because the images in the problem image have the same scale like the target we analyze in this example the rotation angles only. So, Fig. 8c shows one transect in the rotation axis of the Fig. 8a. We can observe three peaks, p1, p2 and p3 and a small peak p4.

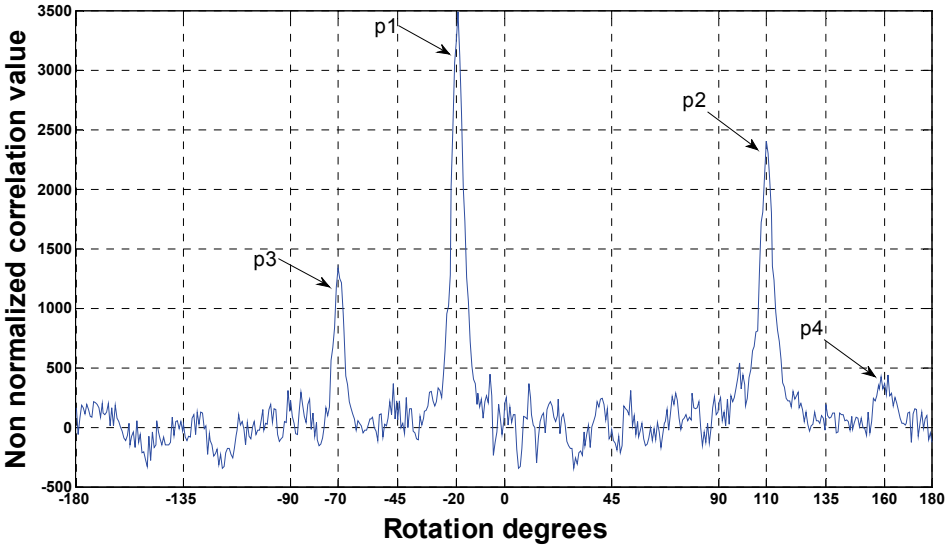
So, in Fig. 8c, the peaks are along the rotation axes that correspond to each letter E rotated in the problem image.



(a)



(b)



(c)

Fig. 8. Invariant correlation to rotation using (a) a nonlinear filter with k=0.3 and PCE=0.0215, (b) a phase only filter with PCE=0.0057, and (c) a rotation transect of (a)

In the image problem of Fig. 8a we have two E letters, one is rotated -20 deg and the other to 110 deg. The peak p1 corresponds to -20 deg, the peak p2 corresponds to 110 deg, the peak p3 to -70 deg and the peak p4 corresponds to 160 deg. Peaks p1 and p4, and peaks p2 and p3

are complementary angles, because when we rotate an object to certain angle, there is another angle for the same object (Fig. 8c). So, in correlation values p1+p4=p2+p3.

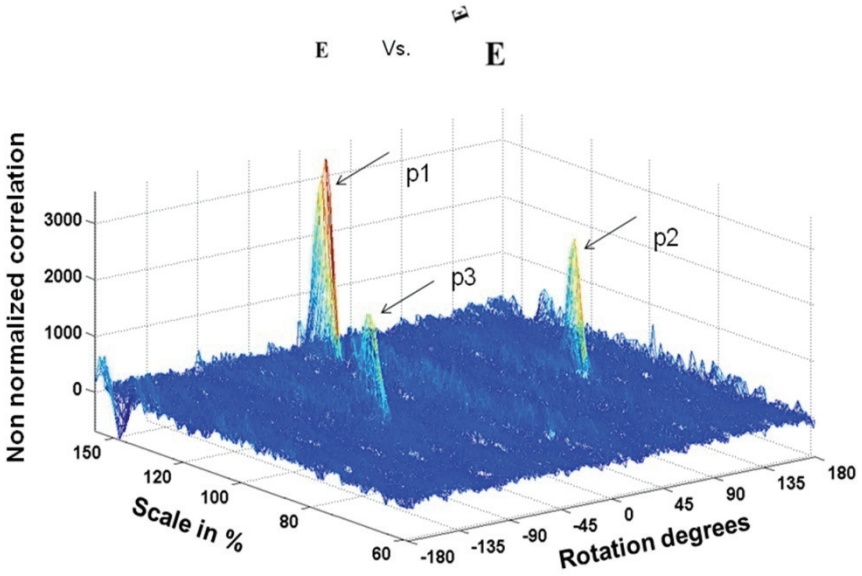
In Fig. 8b we have the appearance of peak p4. This peak corresponds to 160 deg that is complementary to -20 deg of the peak p1. So, p1+p4=p2+p3 again.

In the image problem of Fig. 9 we have two E letters; one is rotated 110 deg and the other to zero deg and scaled to 150% with respect to the filter. The peak p1 corresponds to zero deg. The peak p2 corresponds to 110 deg and its correspondent peak p3 to -70 deg.

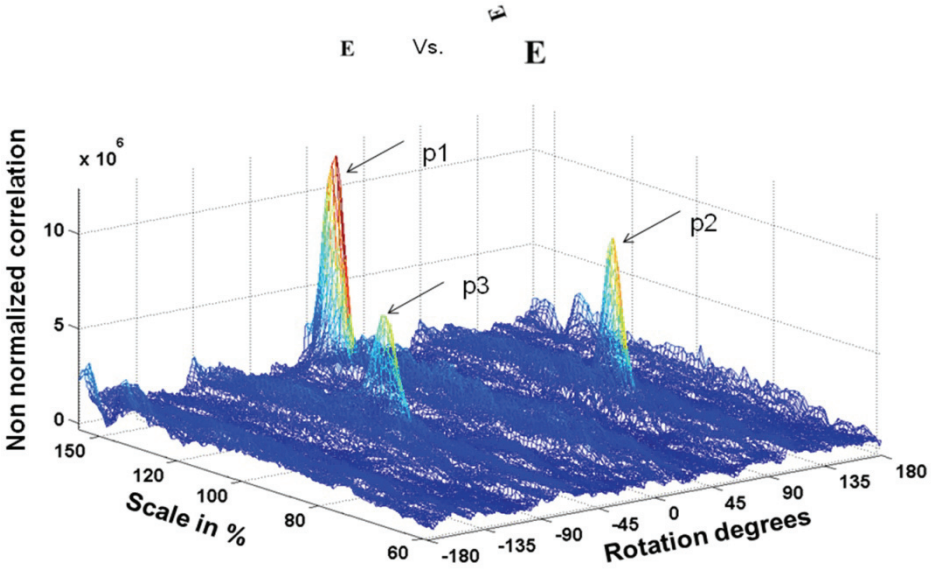
We can observe from these figures that when the rotation angle of the character is small the secondary peak tends to be smaller.

Figs. 9a and 9b show the output correlation plane when we have in the problem image one character with different rotation and scale and we are using a nonlinear and a linear filter respectively in order to make the recognition of the letter. However, one transect along the scale axis is shown in Fig. 9c. The peak p1 is observed exactly in the 150% value in the scale axis.

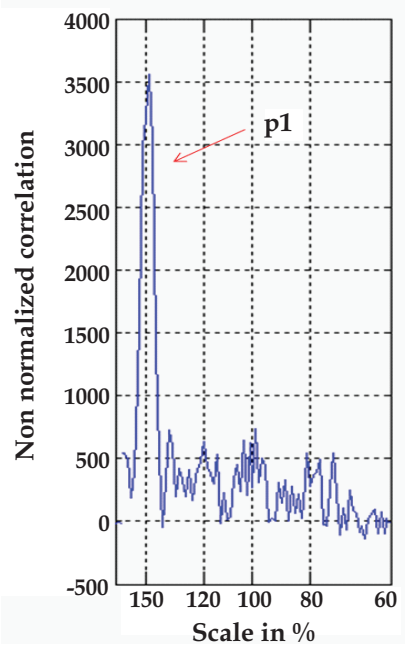
Because we are studying the behavior of the nonlinear versus the linear filter a quantitative comparison between these filters were done using the PCE metric. A desirable attribute of a correlator is capable of producing sharp correlation peaks. This characterization is taken account by the PCE. Considering a nonlinear filter compared with the phase only, the first one concentrates most of the energy that passes through the filter in the correlation peak. We denote PCE_{NLF}, PCE_{POF} the PCE values for the nonlinear filter and the phase only filter, respectively. In this manner, the ratios of these filters were, from Fig. 8: $PCE_{NLF}/PCE_{POF}=3.77$; and from Fig. 9: $PCE_{NL}/PCE_{POF}=4.1$. In all these cases, the PCE value for the nonlinear filter was better compared with the phase only filter.



(a)



(b)



(c)

Fig. 9. Rotation and scale invariant correlation using (a) nonlinear filter with k=0.3 and PCE=0.025, (b) a phase only filter with PCE=0.0061, and (c) a scale transect of (a)

6. Target with noise

In this section we treat with two kinds of noise applied to the target. One is the additive Gaussian noise and the other is the S&PP noise (Fig. 10).

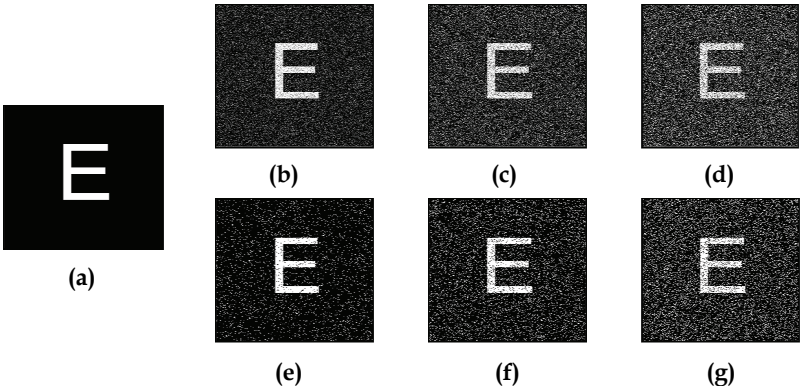


Fig. 10. Target free of noise (a), with additive Gaussian noise of mean zero and variance of 0.1 (b), 0.2 (c) and 0.3 (d), and with S&PP noise of noise density of 0.1 (e), 0.2 (f) and 0.3 (g)

Figs. 11 and 12 show the graphs of the discrimination coefficient using 30 numerical experiments to a 95% level of confidence for a nonlinear filter with k=0.3 when an additive Gaussian noise, with mean zero, and a S&PP noise are considered. In both graphs we have the DC mean *vs.* the variance and the density noise. From these graphs we observe that the filter can recognize the object with a noise variance of approximately 0.27 and a noise density of approximately 0.3 for additive Gaussian and S&PP noise, respectively.

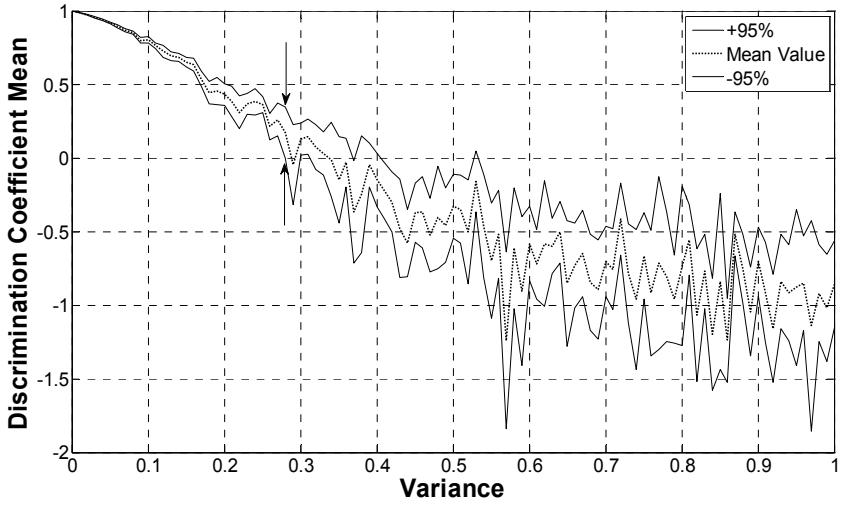


Fig. 11. Performance of a nonlinear filter with k=0.3 in the presence of additive Gaussian noise

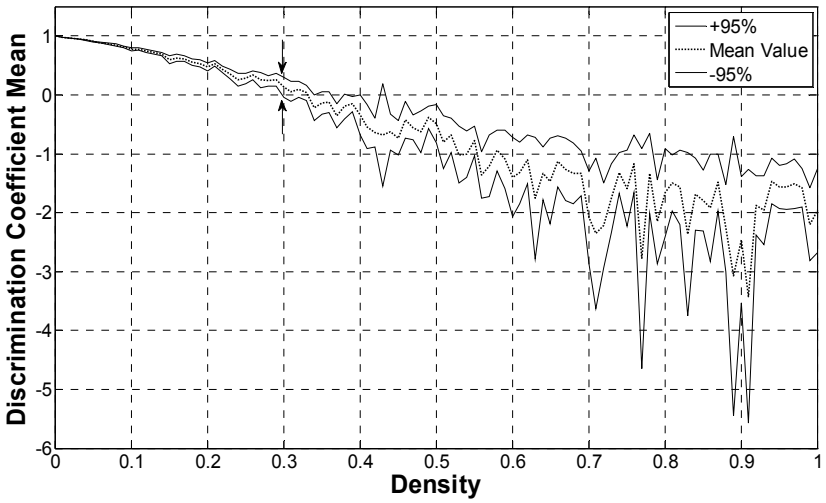


Fig. 12. Performance of a nonlinear filter with k=0.3 in the presence of S&PP noise

It is important to say that the maximum value that can be obtained with the DC is unity, while values below zero indicate that the filter does not recognize the object.

7. DISNF and the Spearman correlation

To study the improvement of the method when the images are embedded in an additive Gaussian noise and in one S&PP noise, we used a nonparametric method, called the rank statistics, to calculate correlation between two images. Substituting the value of each pixel for its corresponding rank, in the normalized correlation expression, a nonlinear correlation expression is obtained [Spearman's rank correlation (SRC)]. Taking this into consideration, the 2-D SRC can be expressed as (Guerrero-Moreno & Álvarez-Borrego, 2009)

$$R(k,l) = 1 - \frac{6\sum_{m,n\in W}(r_t(m,n)-r_s(m+k,n+l))^2}{|W|\cdot(|W|^2-1)}, \tag{9}$$

where $\{r_t(m,n),\ m = 1,2,\ldots N;\ n = 1,2,\ldots M\}$ is the rank of the target, $\{r_s(m+k,n+l),\ k = 1,2,\ldots N;\ l = 1,2,\ldots M\}$ is the rank of the problem image, W is the size of each image, R is the Spearman rank correlation and m, n, k, l are the spatial coordinates.

Fig. 13 shows a block diagram representing the SRC in conjunction with the nonlinear method (SDISNF) in order to increase the tolerance to the noise. In this figure, from the step (1) to the step (10), the procedure is the same as in Fig. 2, except that now, step (11), we realize the inverse FFT for the steps (9) and (10). The step (12) is the SRC in the spatial domain and the step (13) give us the output correlation plane.

From Figs. 14 and 15 we observe the improvement of the filter performance when an additive Gaussian noise, with mean zero, and S&PP noise is added respectively. From these graphs we observe that the filter can support a noise variance of approximately 0.92 and a noise density of approximately 0.54 for additive Gaussian and S&PP noise, respectively.

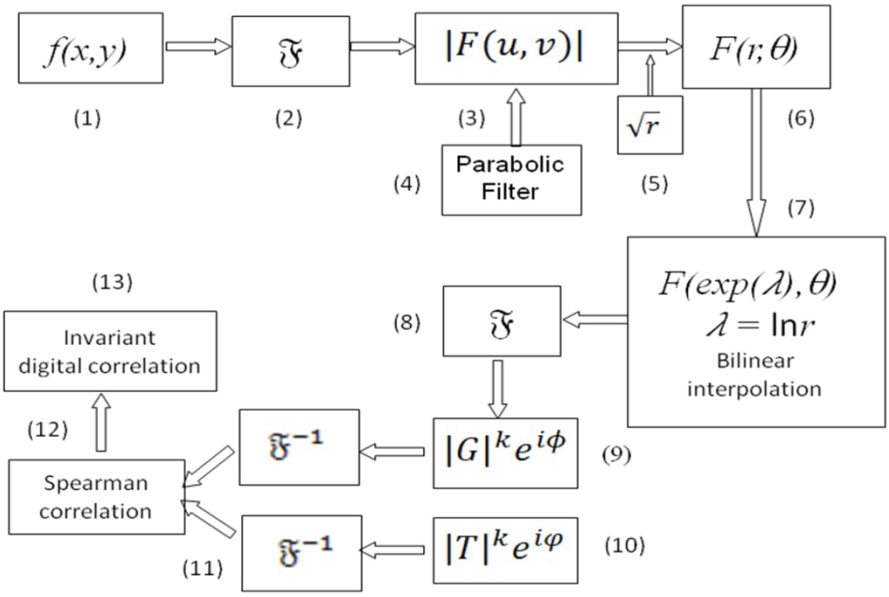


Fig. 13. Simplified block diagram representing the invariant correlation system with a nonlinear filter using the SDISNF

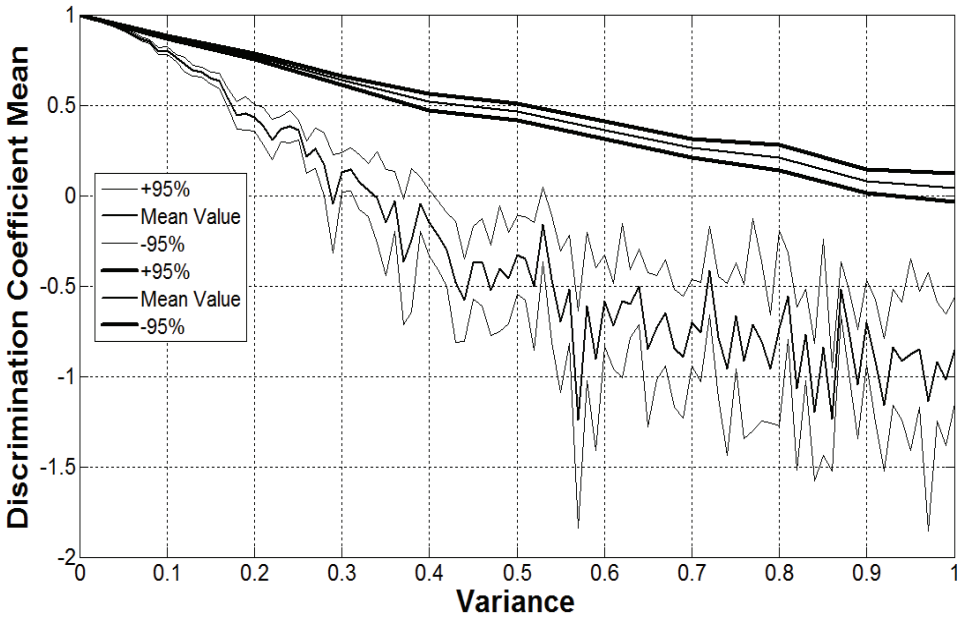


Fig. 14. Performance comparison with SDISNF when the target is embedded in additive Gaussian noise

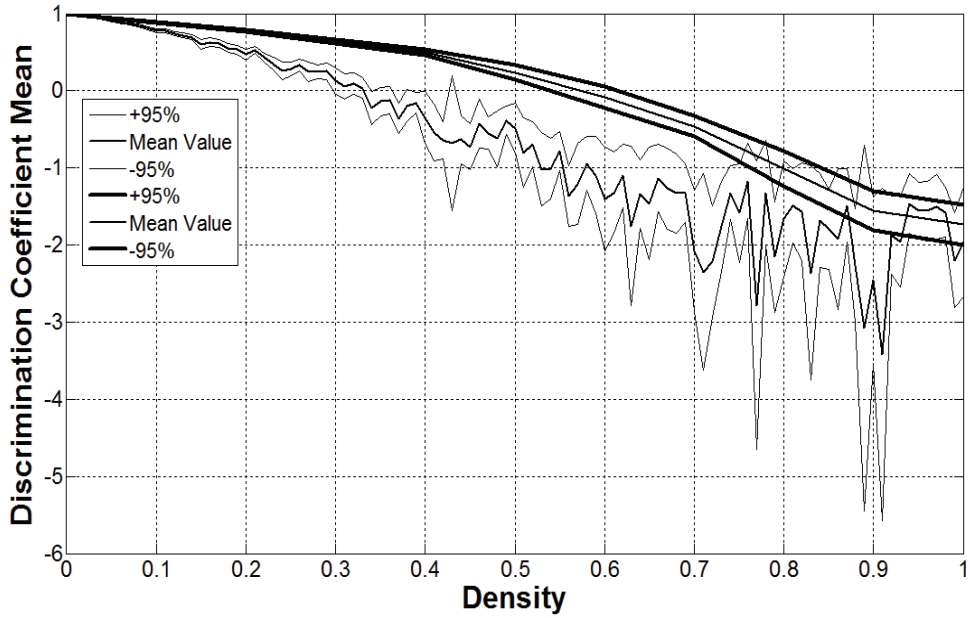


Fig. 15. Performance comparison with SDISNF when the target is immersed in S&PP noise

These figures show the performance comparison when we used the SDISNF for the target immersed in additive Gaussian and S&PP noise. From these figures we can see that the discrimination coefficient is greater than for the only the additive Gaussian noise by a factor of three and for S&PP noise the factor is approximately two.

8. Comparative analysis between different font types and styles letters using a nonlinear invariant digital correlation

In this section we present a comparative analysis of the letters in Times New Roman (TNR), Courier New (CN) and Arial (Ar) font types in Plain & Italicized style and the effects of 5 foreground/background color combinations using an invariant digital correlation system with a nonlinear filter with k=0.3 (Coronel-Beltrán & Álvarez-Borrego, 2010). The evaluation of the output plane with this filter is given by the peak-to-correlation energy (PCE) metric. The results show that the letters in TNR font have a better mean PCE value when is compared with the CN and Ar fonts. This result is in agreement with some studies (Hill, 1997) about text legibility and for readability where the reaction time (RT) of some participant individuals reading a text is measured. We conclude that the PCE metric is proportional to 1/RT.

Fig. 16 shows the mean PCE values obtained versus the letters in Arial (Ar), Courier New (CN) and Times New Roman (TNR) font types in Italicized and Plain style with 5 foreground/background color combinations; black-on-white (BK/W), green-on-yellow (GN/Y), red-on-green (R/GN), white-on-blue (W/BL) and yellow-on-blue (Y/BL), using an invariant digital correlation system with a nonlinear filter with k=0.3. The

foreground/background colors were selected with color coordinates RGB(r,g,b), normalized to [0-255](bytes), as follows; red: (255,0,0), green: (0,128,0), blue: (0,0,255), yellow: (255,255,0), black: (0,0,0) and white: (255,255,255). The letters are 512x512 pixels in size and bitmap file format with 256 colors. The results show that the letters in TNR font have a better mean PCE when is compared with the Ar and CN fonts.

For the italicized style we can observe from this graph that TNR font had a greater mean PCE value compared with Ar and CN fonts. All of them in this style had the same performance and the color combinations were irrelevant.

For the plain style we can observe that TNR font had also a greater mean PCE value compared with Ar and CN fonts. But contrary to the italicized style, in the plain style we have found that the color combinations affect the mean PCE value of the font type letters. The black-on-white and the white-on-blue color combinations had the higher mean PCE values. For the green-on-yellow, red-on-green and yellow-on-blue color combinations, there were no significant changes in the mean PCE values, and like in the italicized case, their behaviour was the same, and the color combinations were irrelevant as well as their mean PCE was statistically not significant.

Using the RT of some participant individuals, reading a text, the effects of six color combinations, three font types (Arial, Courier New & Times New Roman) in Italicized and Plain styles were studied (Hill, 1997) and the results showed that for green-on-yellow, in italicized style, the TNR font had best RT values than Arial font. For all color combinations in italicized and plain style, the TNR font has RT smaller than the Ar font, except for black-on-white and red-on-green in italicized style, and for black-on-white and white-on-blue in plain style.

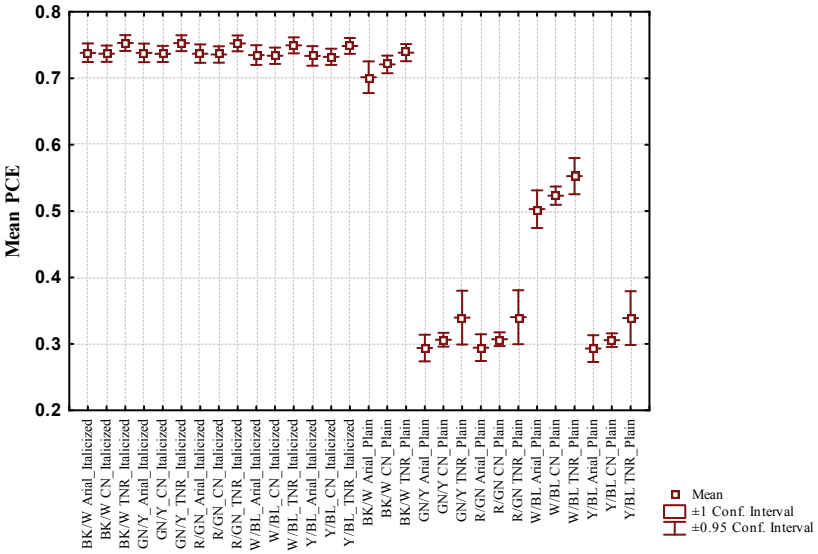


Fig. 16. Performance of the letters Arial (Ar), Courier New (CN) and Times New Roman (TNR) with font types in Italicized & Plain style and 5 foreground/background color combinations

This algorithm was compared with respect to the one published by Alvarez-Borrego & Castro-Longoria (2003), both similar in their use for pattern recognition. Our new algorithm has a computing time that is 20% less. To perform the simulations, a computer, Dell PS_420 with Intel (R) Corel (TM) 2 Quad CPU Q6700 @ 2.66 GHz processor, 2.00 GB of RAM, was used.

9. Conclusion

A digital system for invariant correlation using a nonlinear filter was tested with the maximum nonlinearity strength factor value k=0.3 that was determined experimentally for both scale and rotation. We used the alphabet letters in Arial font type where each one of these letters was taken as a target and correlated with each one for the other letters and showed that our system worked efficiently for the discrimination objects. This nonlinear invariance correlation method was applied using nonlinear and only phase filters for different objects scaled and rotated, Times New Roman font type E letter, and we found a better PCE performance for the nonlinear filters. The DC performance of the noisy target, for additive Gaussian noise and impulse salt-and-pepper noise, was improved significantly applying the SDISNF.

This new combination (SDISNF) is so more powerful than other methods when images with noise are analyzed.

In addition, we presented an analytical method, using a digital invariant correlation, for comparison and to analyze different font and style letters with five color combinations. Our results showed a better output for the TNR in comparison with the Ar font letters in italicized & plain style. And generally, italicized letter style had greater PCE values than plain style letter. These results were in accord to some other studies realized with subjective methods. Our results can be useful for studying the effects of background color combinations on legibility displayed on a computer screen from the Web, email, or other texts written in books, newspapers, magazines, etc., and its improvement as well as in readability. We conclude that the PCE metric is proportional to 1/RT. It would be interesting investigate the relationship between our method based in analytical mathematical functions and those other subjective empirical methods used by psychologists and Web designers, among others.

10. Acknowledgment

This work was partially supported by the grant 102007 from CONACYT and a grant from Universidad de Sonora.

11. References

Álvarez-Borrego, J. & Castro-Longoria, E. Discrimination between *Acartia* (Copepoda: Calanoida) species using their diffraction pattern in a position, rotation invariant digital correlation. *J. Plankton Res.,* Vol. 25, No. 2, pp. 229-233, ISSN 0142-7873

Casasent, D. & Psaltis, D. (1976a). Scale invariant optical correlation using Mellin transforms. *Opt. Commun.,* Vol. 17, No. 1, April 1976, pp. 59-63, ISSN 0030-4018

Casasent, D. & Psaltis, D. (1976b). Scale invariant optical transform. *Opt. Eng.,* Vol. 15, No. 3, pp. 258-261, May-June 1976, ISSN 0091-3286

Casasent, D. & Psaltis, D. (1976c). Position, rotation, and scale invariant otical correlation. *Appl. Opt.,* Vol. 15, No. 7, July1, pp. 1795-1799, ISSN 0003-6935

Cohen, L. (1993). The scale representation. *IEEE-SP,* Vol. 41, No. 12, Dec. 1993, pp. 3275-3292, ISSN 1053-587X

Cohen, L. (1995). *Time Frequency Analysis* (A.V. Oppenheim), Prentice Hall, ISBN-10 0135945321, New Jersey

Coronel-Beltrán, A. & Álvarez-Borrego, J. (2008). Nonlinear filter for pattern recognition using the scale transform, *Proc. SPIE,* Vol. 7073, 70732H, San Diego, California, USA, August 2008

Coronel-Beltrán, A. & Álvarez-Borrego, J. (2010). Comparative analysis between different font types and letter styles using a nonlinear invariant digital correlation. *J Mod Optics,*Vol. 57, No. 1, (January 2010), pp. 58-64, ISSN 0950-0340 print/ISSN 1362-3044 online

Cristóbal, G.; Cuesta, J. & Cohen, L. (1998). Image filtering and denoising through the scale transform, *Proc. IEEE-SP, International Symposium on Time-Frequency and Time-Scale Analysis,* ISBN 0-7803-5073-1, pp. 617-620, Pittsburgh, PA, USA, October 1998

De Sena, A. & Rocchesso, D. (2004). A fast Mellin transform with applications in DAFx, *Proceedings of the 7th International Conference on Digital Audio Effects (DAFx'04),* Naples, Italy, October 2004, pp. 65-69

De Sena, A. & Rocchesso, D. (2007). A Fast Mellin and Scale Transform. *EURASIP Journal on Advances in Signal Processing,* Vol. 2007, pp. 1-9

Goodman, J.W. (2005). *Introduction to Fourier Optics*, Roberts & Company, ISBN 0-9747077-2-4, Greenwood Village, CO, USA, pp. 435-436

Guerrero-Moreno, R. & Álvarez-Borrego, J. (2009). Nonlinear composite filter performance. *Opt. Eng.* Vol. 48, No. 6, 067201, June 2009, ISSN 0091-3286

Hill, A.L. (1997). Readability of websites with various foreground/background color combinations, font types and word styles. Stephen F. Austin State University. http://www.laurenscharff.com/research/AHNCUR.html

Horner, J.L. & Gianino, P.D. (1984). Phase-only matched filtering. *Appl. Opt.* Vol. 23, No. 6, March 15, pp. 812-816, ISSN 0003-6935

Javidi, B. (1989a). Nonlinear joint power spectrum based optical correlation. *Appl. Opt.* Vol. 28, No. 12, June 15, pp. 2358-2367, ISSN 0003-6935

Javidi, B. (1989b). Synthetic discriminant function-based binary nonlinear optical correlator. *Appl. Opt.* Vol. 28, No. 13, July 1, pp. 2490-2493, ISSN 0003-6935

Javivi, B. (1989c). Nonlinear matched filter based optical correlation. *Appl. Opt.* Vol. 28, No. 21, November 1, pp. 4518-4520, ISSN 0003-6935

Javidi, B. (1990). Comparison on nonlinear joint transform correlator and nonlinear matched filter based correlator. *Opt. Commun.* Vol. 75, No. 1, 1 February 1990, pp. 8-13, ISSN 0030-4018

Javidi, B. & Horner, J.L. (1994). *Real-Time Optical Information Processing*, Academic Press, Boston, ISBN 0123811805

Pech-Pacheco, J.L.; Cristóbal, G., Álvarez-Borrego, J. & Cohen, L. (2000). Power cepstral image analysis through the scale transform, *Proc. SPIE*, Vol. 4113, pp. 68-79

Pech-Pacheco, J.L.; Cristóbal, G., Álvarez-Borrego, J. & Cohen, L. (2001). Automatic system for phytoplanktonic algae identification. *Limnetica,* Vol. 20(1), pp. 143-157, ISSN 0213-8409

Pech-Pacheco, J.L.; Álvarez-Borrego, J., Cristóbal, G. & Mathias, S.K. (2003). Automatic object identification irrespective to geometric changes. *Opt. Eng.*, Vol. 42, No. 2, February 2003, pp. 551-559, ISSN 0091-3286

Schwartz, E.L. (1977). Afferent Geometry in the Primate Visual Cortex and the Generation of Neuronal Trigger Features. *Biol. Cybernetics,* Vol. 28, No. 1, pp. 1-14, ISSN 0340-1200

Schwartz, E.L. (1980). Computational anatomy and functional architecture of striate cortex: A spatial mapping approach to perceptual coding. *Vision Res.,* Vol. 20, No. 8, Aug. 1980, pp. 645-669, ISSN 0042-6989

Schwartz, E.L.; Christman, D. R. & Wolf, A.P. (1984). Human primary visual cortex topography imaged via positron tomography. *Brain Res.,* Vol. 294, No. 2, Mar. 1984, pp. 225-230, ISSN 00068993

Vander Lugt, A.B. (1964). Signal detection by complex spatial filtering. *IEEE Trans. Inf. Theory,* Vol. 10, No. 2, Apr. 1964, pp. 139-145, ISSN 0018-9448

Vijaya Kumar, B.V.K. & Hassebrook, L. (1990). Performance measures for correlation filters. *Appl. Opt.,* Vol. 29, No. 20, July 10, pp. 2997-3006, ISSN 0003-6935

13

Pattern Recognition of Digital Images by One-Dimensional Signatures

Selene Solorza[1],
Josué Álvarez-Borrego[2] and Gildardo Chaparro-Magallanez[2]
[1]*UABC, Facultad de Ciencias*
[2]*CICESE, División de Física Aplicada, Departamento de Óptica*
México

1. Introduction

Since the computer's evolution in the middle of last century, pattern recognition of digital images based on correlations has been applied in science as well as technology areas. Their applications are broad and variety(Gonzalez & Woods, 2002). The techniques developed are used to identify micro-objects, for example the inclusion of virus bodies, bacteria, chromosomes, etc. (Álvarez-Borrego & Castro-Longoria, 2003; Álvarez-Borrego et al., 2002; Álvarez-Borrego & Solorza, 2010; Bueno et al., 2011; Forero-Vargas et al., 2003; Pech-Pacheco & Álvarez-Borrego, 1998; Zavala & Álvarez-Borrego, 1997). The analysis of those samples requires experience and moreover, the samples analyzed frequently contain material with different fragmentation degrees and this can lead to confusion and loss of information. There are other works in which the pattern recognition are based on probabilistic methodologies; here the objects are represented by their statistical characteristic features (Fergus et al., 2003; Holub et al., 2005). These are used to face identification and image restoration (Alon et al., 2009; Kong et al., 2010a;b; Ponce et al., 2006); fingerprints classification (Jain & Feng, 2011; Komarinski et al., 2005; Moses et al., 2009). Also, these works are useful to classify or count micro-objects, where the object variation, background, scale, illumination, etc., are taking into account (Arandjelovic & Zisserman, 2010; Barinova et al., 2010; Lempitsky & Zisserman, 2010).

Recently, digital systems invariant correlation to position, rotation and scale are utilized in the pattern recognition field (Álvarez-Borrego & Solorza, 2010; Coronel-Beltrán & Álvarez-Borrego, 2010; Lerma-Aragón & Álvarez-Borrego, 2009a;b; Solorza & Álvarez-Borrego, 2010). Such invariants are made of by the Fourier and Fourier-Mellin transforms in conjunction with non linear filters (*k*-law filter). The non linear filters have advantages compared with the classical filters (POF, BAPOF, VanderLugt, CHF) due to their great capacity to discriminate objects, the maximum value of the correlation peak is well localized, and the output plane is less noisy (Guerrero-Moreno & Álvarez-Borrego, 2009). At the same decade, different contributions have been proposed to identify the target when different kinds of noise were presented in the input scene, one of the techniques that had a very good performance is the adaptive synthetic discriminant functions (ASDF) (González-Fraga et al., 2006) also used in an optical system (Díaz-Ramírez et al., 2006).

There are many works about digital systems invariant to position, rotation and scale but they have a big computational cost. The composite non linear filters like ASDF filters can be used for obtain a digital invariant correlation to rotation, but they must have the information of all angles of rotation of the object to be recognized. In order to have a new correlation digital system invariant to position, rotation and scale with low computational cost, in this work a new methodology based on one-dimensional signatures of the images is presented.

Other important aspect in the study of digital system is the information leak. Then, the objective is neglect only the irrelevant information when the mask applies. Hence, we used binary rings masks associated to the image. The mask is constructed based on the real part of the Fourier transform of a given image, therefore each image will have one unique binary rings mask, called adaptive mask, in this form the information leak is avoided because the frequencies filtered are not the same always (Solorza & Álvarez-Borrego, 2010).

This chapter presents three digital systems based on the Fourier plane, adaptive binary rings masks, one-dimensional signatures and non linear correlations. Section 2 presents the methodology to obtain the adaptive binary rings mask of the image, the technique to associate the signature at each image and the non linear correlation used in the classification. Also, a position and rotation invariance digital system is presented and tested in the classification of fingerprints digital images. This section shows that the modulus of the Fourier transform of the image is the natural and easier manner to achieve the invariance translation in the digital system; taking advantage of the translation invariant property of the Fourier transform. Also, presents how the rotational invariance is obtained by the binary rings masks filters. In section 3, a digital system invariance to position, rotation and scale is built and used to classify image of Arial font type which are shifted, scaled and rotated. Here, the scale transform (which is a particular case of the Mellin transform) is introduced and utilized for the scale recognition. Section 4, gives an alternative method to get scale invariance using composite filters. Finally, concluding remarks are presented.

2. The digital system

An important aspect in the study of digital systems is the information leak. In this work, the frequencies are filtered using binary rings masks and the objective is neglect only the irrelevant information when the mask applies. Therefore, it is proposed a methodology to build a mask associated to the image, in this form, the information leak is avoided because the frequencies filtered are not the same always. Next, the procedure to construct a binary rings mask of the image chosen is explained.

2.1 Binary rings mask of the image

The mask associated of a given image, named I, is built by taking the real part of its $2D$-fast Fourier transform (FFT), given by

$$f(x,y) = Re(FFT(I(x,y))), \tag{1}$$

where (x,y) represent a pixel of the image. For example, Fig. 1(b) shows the real part of the fast Fourier transform (Eq. (1)) of the 417×417 gray-scale image in Fig. 1(a). This $3D$-graph is $2D$-mapping by $\left[min_{1 \leq x,y \leq 417}\{f(x,y)\} \quad max_{1 \leq x,y \leq 417}\{f(x,y)\} \right] \rightarrow \left[0 \quad 1 \right]$, shown in Fig. 1(c).

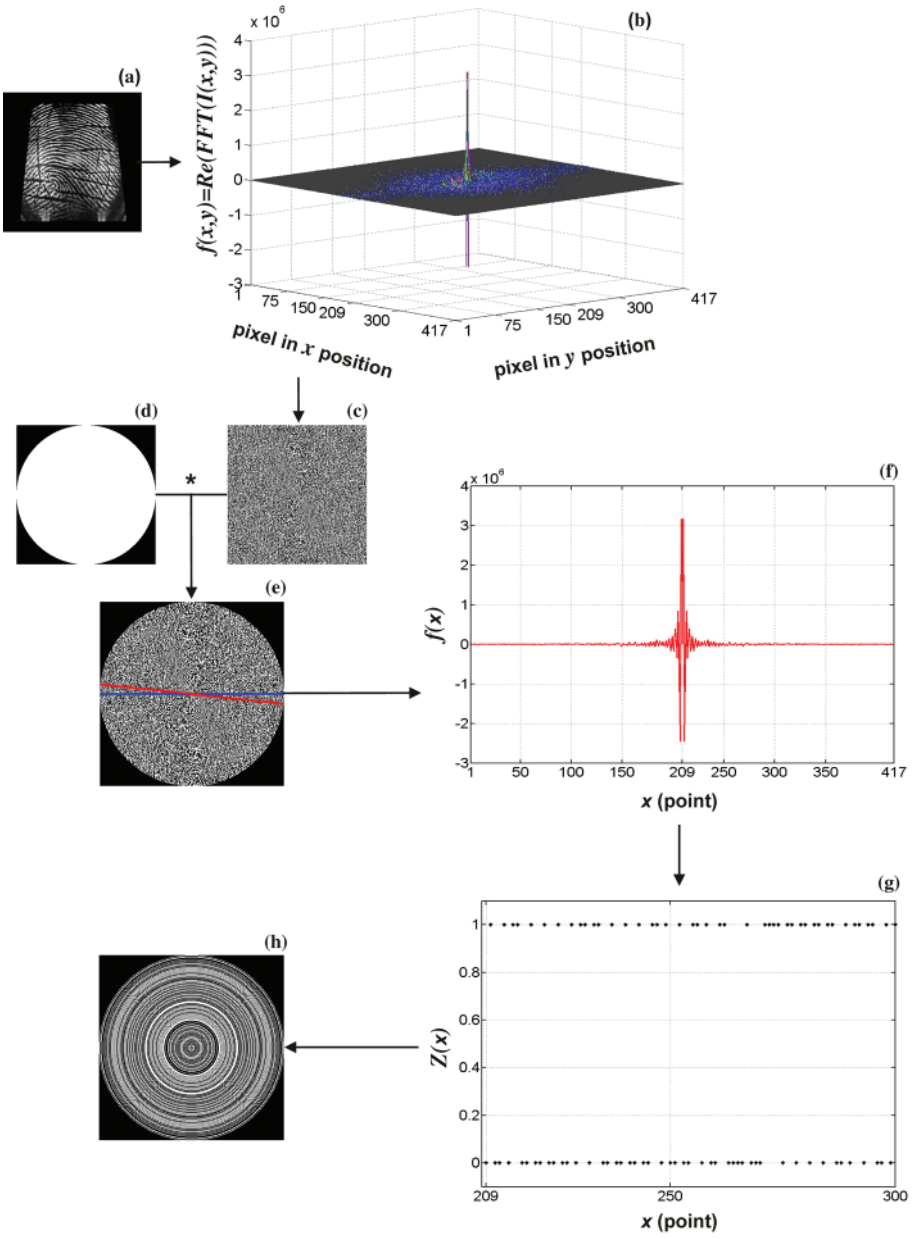


Fig. 1. Binary rings mask construction example. The asterisk means the point to point multiplication of both images.

Next, this result is filtered by the binary disk of 417 pixels diameter presented in Fig. 1(d). Based on the result of this operation, we obtain 180 profiles of 417 pixel length that passes for the $(209, 209)$ pixel, center of the image (separated by a $\Delta\theta = 1$ deg, sampling of this way the entire circle). The next step is to compute the addition of the intensity values in each profile and after that select the profile whose sum has the maximum value, it is called the maximum intensity profile. Fig. 1(e), displays the profile of zeroth degrees (blue-line) and the maximum intensity profile in red-line ($\theta = 174$ degrees). Fig. 1(f) shows the graph of the maximum intensity profile in the cartesian plane, where $\mathbf{x} = 1, 2, 3, \ldots, 417$ represents the points $(r\cos\theta, r\sin\theta)$, $-209 \leq r \leq 209$ and $\theta = 174$. Notice the symmetry of the graph in the $\mathbf{x} = 209$ axis, which will be preserved in the one-variable binary function

$$Z(\mathbf{x}) = \begin{cases} 1, & f(\mathbf{x}) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The graph of the Z function is plotted in Fig. 1(g), for clarity and due to the symmetry of the Z function, we only plot the points $209 \leq \mathbf{x} \leq 300$. Next, taking $\mathbf{x} = 209$ as the rotation axis, the graph of Z is rotated 180 degrees to obtain concentric cylinders of height one, different widths and centered in $(209, 209)$. Finally, mapping those cylinders in two dimensions we built the binary rings mask associated to the image. The mask for the fingerprint digital images given in Fig. 1(a) is shown in Fig. 1(h).

Each fingerprint digital image used in this work (Fig. 2) has its own binary rings mask Fig. 3. Notice how the mask changes with the images, so they are called adaptive mask. The next step in the construction of the digital system is build a signature for the image.

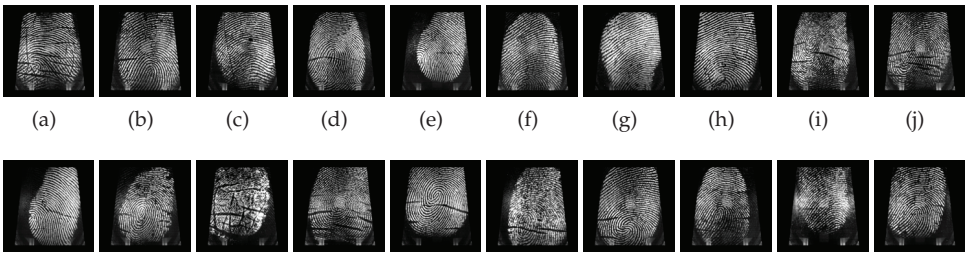


Fig. 2. Fingerprints digital images. The images was downloaded from Biometrics Ideal Test web page (biometrics.idealtest.org).

2.2 The signature of the image

The aim of this work is identify a specific target (the object to be recognized) no matter the position or the angle of rotation presented on the plane. For example, using the fingerprint image in Fig. 4(a), the invariance to position is obtained using the modulus of the frequency content of the image, $|FFT(I)|$ (Fig. 4(b)). Next, the mask (Fig. 4(c)) is applied in the Fourier plane for sampling the frequencies pattern of the object (Fig. 4(d)). Finally, the different modulus values of the Fourier transform for each ring is summed and then assigned to the

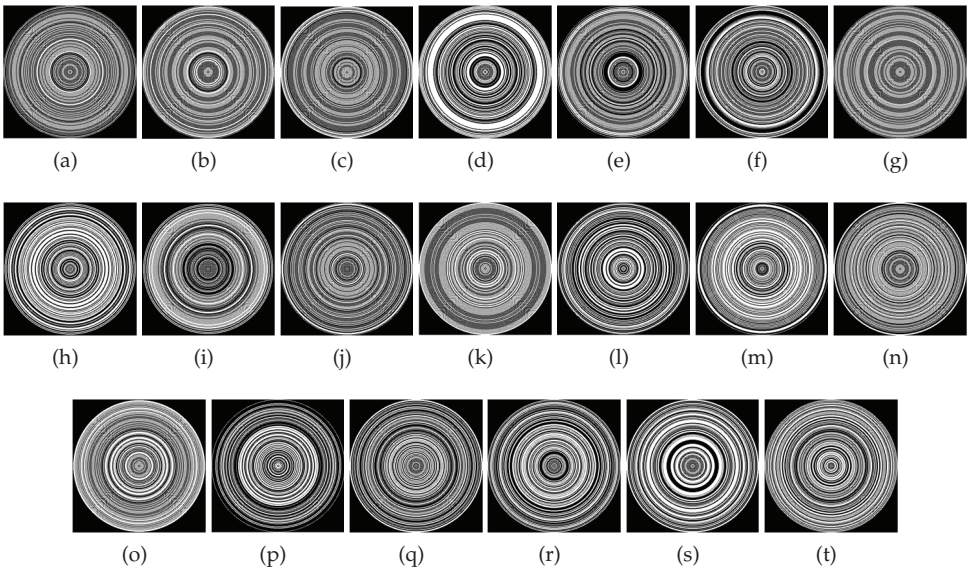


Fig. 3. Binary masks associated at images in Fig. 2.

corresponding ring index to obtain the signature of the image (Fig. 4(e)). Here, the rings are numbered from the center to outside of the circles.

In the target images data base we have twenty image without rotation (Fig. 2), thus we calculated twenty different binary rings masks (Fig. 3). Each mask will be applied at Fig. 4(b) to obtain twenty different signatures, which are shown in Fig. 5(a) in black-dashed curves. Because of the use of different rings masks the length of the signatures are different, hence zeros are added at the end of those signatures shorter than the length of the larger signature to match the size of all of them. Finally, taking the average of those twenty signatures the average signature of the image is displayed as the red-curve in Fig. 5(a). This procedure has the advantage that more information is incorporated in the average signature because of the relevant information filtered by the twenty different masks used instead of the information obtained from one mask only. When the image is changed, it is expected that the average signature changes. But, because of the uses of the adaptive rings masks associated to the target images data base, the hypothesis is that when the image presents a rotation, the average signature will be similar to the average signature of the non-rotated image, hence the digital system will has the capability to identify rotated objects. Fig. 5(b) shows the average signature of the Fig. 2(a) without rotation (called a_0) and with a rotation angle of 10 degrees (a_{10}), also the average signature of Fig. 2(b) is presented (b_0). As it is expected the form of the average signatures a_0 and a_{10} are very similar when the signature has high values, not being the case for the average signature of b_0. Notice that the construction of the signature is related to the adaptive binary rings masks, hence if the images resolution changes the masks also change, this could be a problem for the digital system that should be take into account, however this topic is beyond this work and hereafter we assume that the images have the same resolution

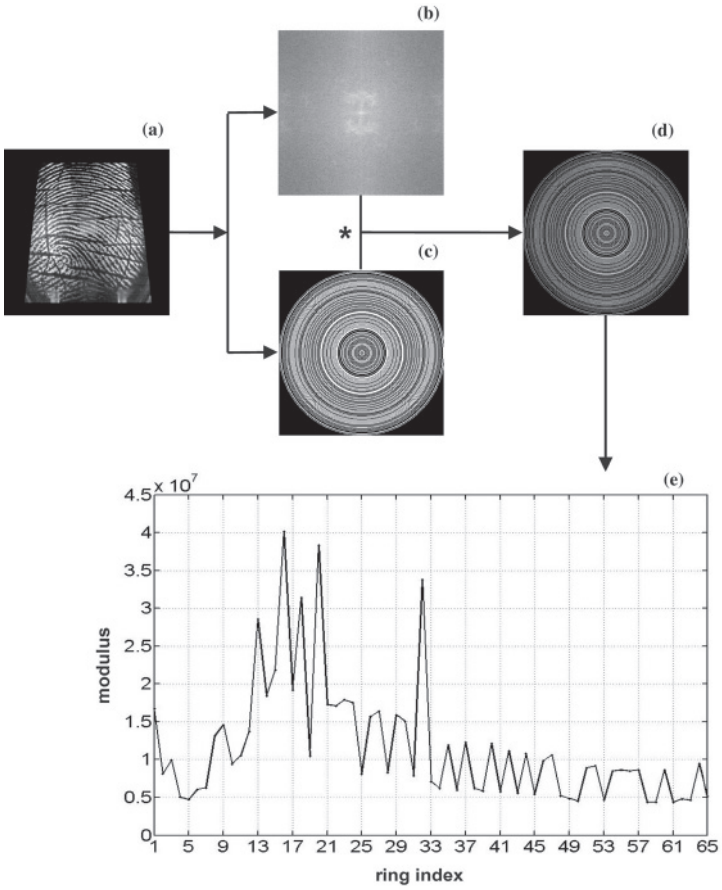


Fig. 4. Signature of a given fingerprint. Only for visualization purposes the (b) and (d) figures are shown in log scale. The asterisk means the point to point multiplication of both images.

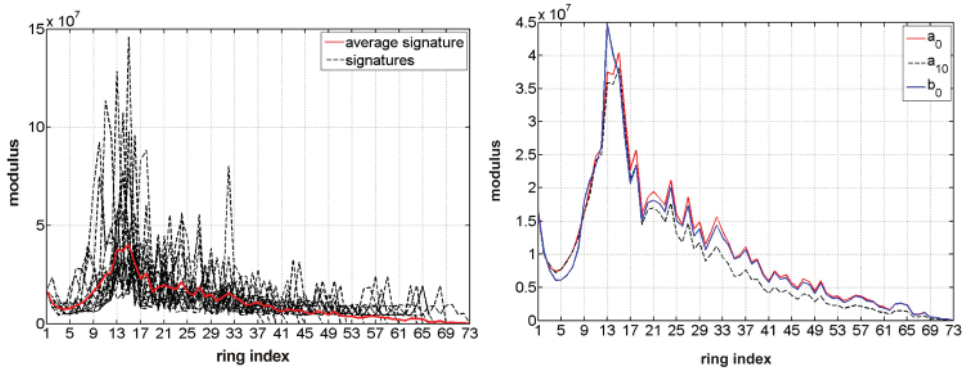
always. Once the average signatures can be assigned at each image, the next step is the classification.

2.3 The non linear correlation

The average signature of the problem image (PI) is compared with the average signature of the target (T) to recognize or not the target using the non linear correlation, C_{NL},

$$C_{NL}(\mathrm{PI},\mathrm{T}) = \mathrm{PI} \otimes \mathrm{T} = FFT^{-1}\left(|FFT(\mathrm{PI})|^k \, e^{i\phi_{PI}} \, |FFT(\mathrm{T})|^k \, e^{-i\phi_T} \right), \tag{3}$$

where $\otimes$ means correlation, $i = \sqrt{-1}$, ϕ_{PI} and ϕ_T are the phases of the fast Fourier transform of the problem image and the target, respectively, $0 < k < 1$ is the non linear coefficient factor (Solorza & Álvarez-Borrego, 2010). Fig. 6(a) shows the result of Eq. (3) with $k = 0.1$ using Fig. 2(a) as the target (its average signature is given in Fig. 5(b) as a_0), and the problem

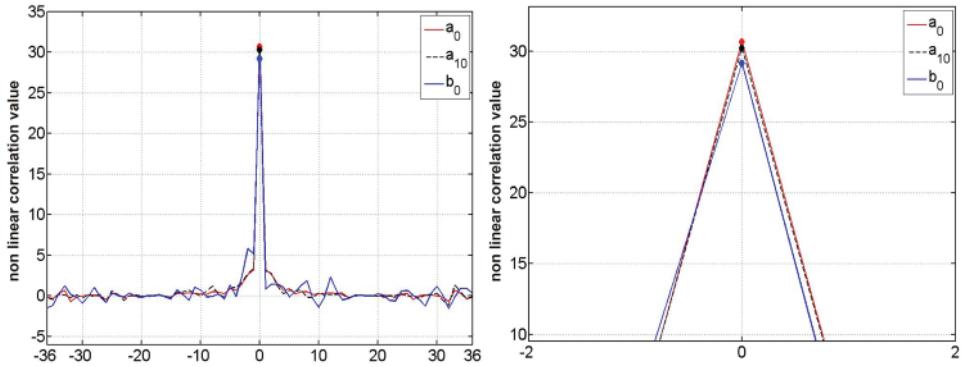


(a) Average signature associated to the fingerprint in Fig 4a.

(b) Some average signature examples.

Fig. 5. Average signatures.

images taken were Fig. 2(a) without rotation (autocorrelation), its rotated ten degrees image (average signature a_{10}), and the Fig. 2(b) (average signature b_0). If the maximum value of the magnitude for the correlations (plotted as dots, Fig. 6(b)) are significant, that is similar to the autocorrelation maximum value, hence the PI contains the target, otherwise has a fingerprint different to the target.



(a) Fig. 2(a) was chosen as target.

(b) Amplification around the center.

Fig. 6. The non linear correlation vector examples.

2.4 The position and rotation invariance digital system applied to the fingerprint data base

The position and rotational invariance non linear correlation digital method described in the algorithm in Fig. 7 (steps explained in above sections) was applied to the twenty 417×417 gray-scale fingerprint digital images (Fig. 2). Each image was rotated ± 15 degrees, one by one, hence in the problem images data base we worked with 620 images, moreover the saw tooth

effect (noise) is incorporated into the problem. Therefore, the digital system is more robust in the pattern recognition. Each image in Fig. 2 was selected as target, thus the target data base has twenty image only and the corresponding average signature was constructed. Those twenty average signatures were correlated with the 620 average signatures of the rotated images using $k = 0.1$ in Eq. (3). The maximum value of the magnitude for the corresponding correlation was assigned to the image. The results were box plotting by the mean correlation with two standard errors ($\pm$ 2SE) and outliers.

It is impossible to show the box plot developed for each target used; hence, to exemplify we choose Fig. 2(a), (b) , (c) and (d) as targets. The mean of the maximum of the magnitude for the non linear correlation values are shown in Fig. 8. In the horizontal-axis are setting the letters corresponding to the images in Fig. 2. The vertical-axis represents the mean of the maximum values without normalize the magnitude of the non linear correlation for the images of each fingerprints and their corresponding rotated images.

The digital system has a confidence level of 95.4% according with the statistical analysis, when the target is the fingerprint in Fig. 2(f). Moreover, when the other images are chosen as target the system gives a confidence level of 100%. Therefore, the digital system in Fig. 7 shows an excellent performance in the identification of fingerprints images.

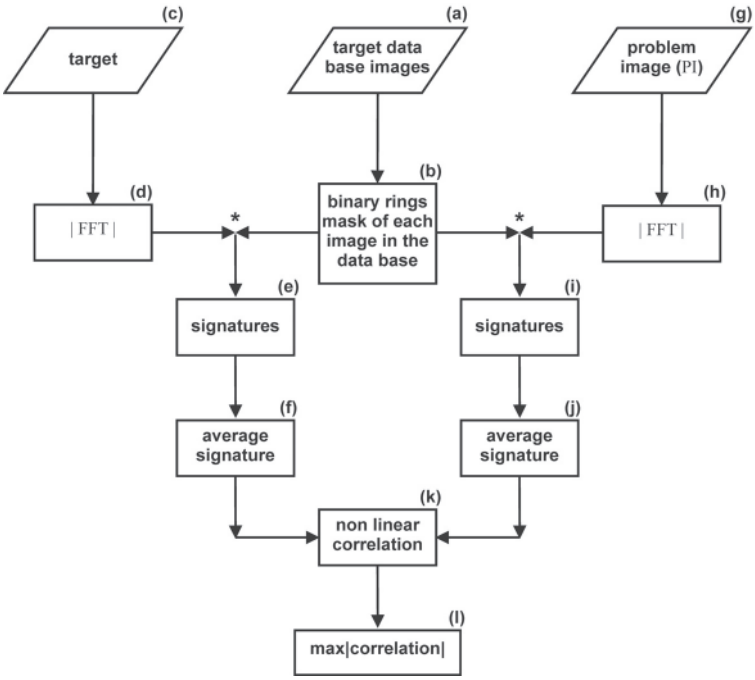


Fig. 7. Position and rotation invariant non linear correlation digital system algorithm. The asterisk means the point to point multiplication of both images.

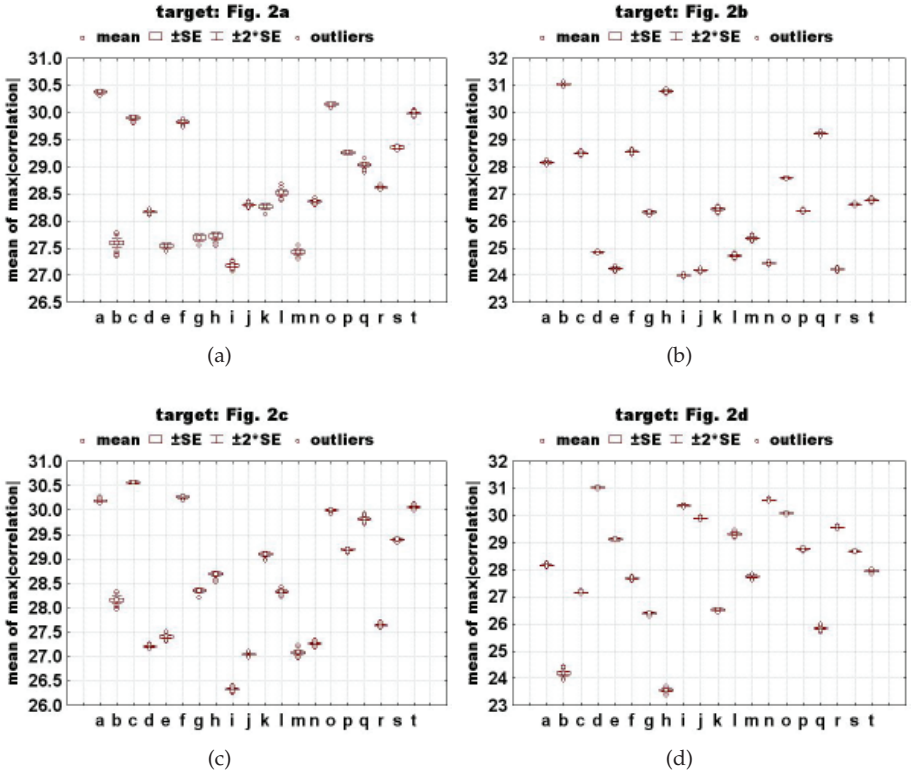


Fig. 8. Box plot examples for some fingerprint images.

3. The position, rotation and scale invariance methodology

In some analysis of real image is so important to consider the scale invariance. In order to analyze this case, images of Fig. 9 were used (hence, the target data base has five image). They are black and white image of 257×257 white contours of Arial font type. We choose this font type and these image because of their similarity between some of them. Each image were scaled $\pm 5\%$, one by one. Hence, we have eleven image for each letter obtaining a total of 55 images. Each of those 55 images were rotated 360 degrees, degree by degree, so we worked with 19,800 in the problem images data base to test the methodology. Fig. 9(f) shows a 257×257 images with the B letter's contour rotated 315 degrees and incremented 5% from its original size (Fig. 9(a)).

To introduce the scale invariance in the methodology described in Fig. 7, the average signatures for the target and the problem image (PI) are computed similar of those in Fig. 7(f) and Fig. 7(j). The difference in this procedure is that the binary rings mask of the problem image (Fig. 10(c)) is also computed and the signature obtained from this mask, Fig. 10(g), contributes to build the average signature for the target too, Fig. 10(h). Figure 10(i) shows an example of the average signature when B letter is used like a target. Analogously, to obtain

Fig. 9. 257 × 257 black and white images with Arial font type letter contours.

the average signature for the PI in Fig. 10(c) the same procedure is following but instead of the $|FFT|$ of the target the $|FFT|$ of the PI is used and the same six binary rings mask are utilized to construct the corresponding average signature. Therefore, when the PI changes the average signature of the target changes too, because the associated mask is different. However, the adaptive mask of the target and the PI are utilized in the construction of their corresponding

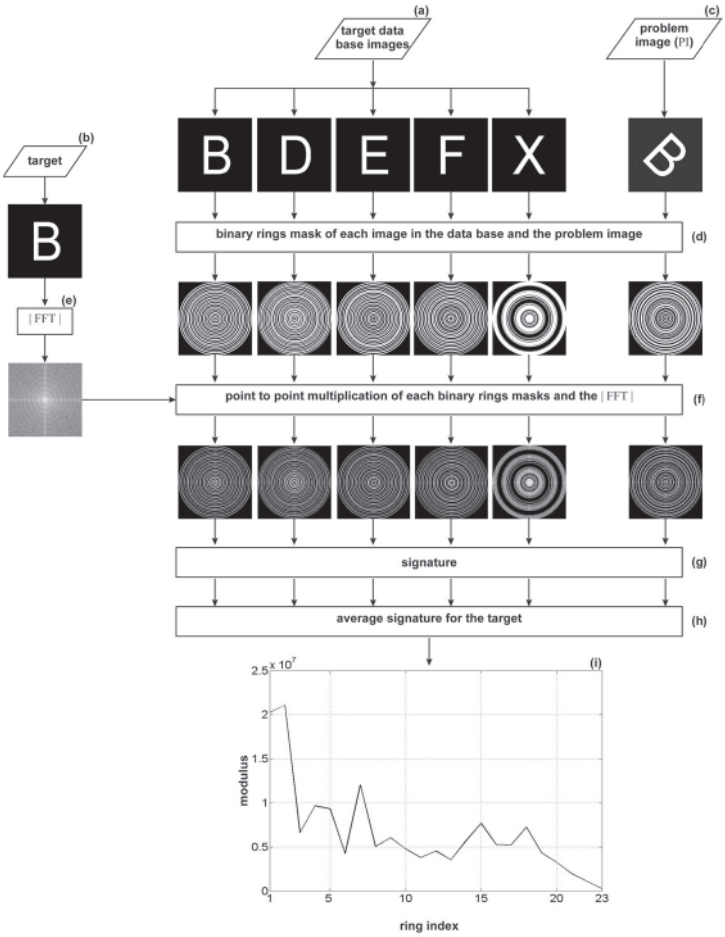


Fig. 10. Position, rotation and scale invariant average signature.

average signature, hence, when the PI is a rotated image of the target the same frequencies are filtered, so the average signature of both image are similar not being the case when the target and the PI have different letters.

Fig. 11 shows the digital system algorithm modified to incorporate scale invariance in the algorithm of Fig. 7. Once we have the average signature for the target (Fig. 11(g)) and PI (Fig. 11(l)), in order to determine if the PI is the rotated and/or scaled target or another image, the scale transform for the average signature of the target, named $D_T(c)$ (Fig. 11(h)) and the problem image, $D_{PI}(c)$ (Fig. 11(m)), are obtained by the scale transform function (De Sena & Rocchesso, 2004)

$$D_s(c) = \frac{1}{\sqrt{2\pi}} \int_0^\infty s(t) e^{(-ic-1/2)\ln t} dt, \tag{4}$$

where $s(t)$ represents the average signature of the image. Then, the $|D_{PI}(c)|$ (Fig. 11(i)) is compared with the $|D_T(c)|$ (Fig. 11(n)) to recognize or not the target using the non linear correlation Eq. (3) with $k = 0.3$. Finally, if the maximum value for the magnitude of the correlation is significant the PI contains the target, otherwise has a different image of the target.

The statistical analysis using $\pm$ 2SE and outliers gives that the digital system has a confidence level of 95.4% when the targets are the image of B and D and a 100% when the targets are E, F and X, (Fig. 12). Hence, the digital system invariant to position, rotation and scale shows an excellent performance.

4. Composite filters for scale invariance

One alternative method to get scale invariance is the use of training images to generate composite filter with Fourier transform too. This methodology have information of the target in different sizes. Fig. 13 shows the procedure to train the composite filter that will be used in the digital system invariance to rotation, scale and position. Once it is established the number of training images, for example I_1, I_2, ..., I_n (Fig. 13(a)), the modulus of the fast Fourier transform of each image is calculated, that is $|FFT(I_1)|$, $|FFT(I_2)|$, ..., $|FFT(I_n)|$ (Fig. 13(c)). Then, the sum of all those modulus is done to generate a new composite Fourier plane (Fig. 13(d)), which has more information than that obtained using only one image. Mathematically, this new Fourier plane is expressed as

$$f_c = \sum_{k=1}^{n} |FFT(I_k)|. \tag{5}$$

The Fourier plane obtained by f_c (Fig. 13(e)) can substitute the modulus plane that is presented in the procedure shown in Fig. 4(b). Then, using the binary rings mask the one-dimensional signature for a given target image is obtained.

The composite filter was applied to the 256 $\times$ 256 gray-scale diatoms images shows in Fig. 14. The diatoms are a class of microscopic unicellular algae, are photosynthetic organisms that live in sea water being a quite important part in the food chain. Each image in Fig. 14 was scaled in the range of 90% to 107% (one by one) and all those image were rotated 360 degrees (every 45 degrees). Hence, we have processed 528 images in this example.

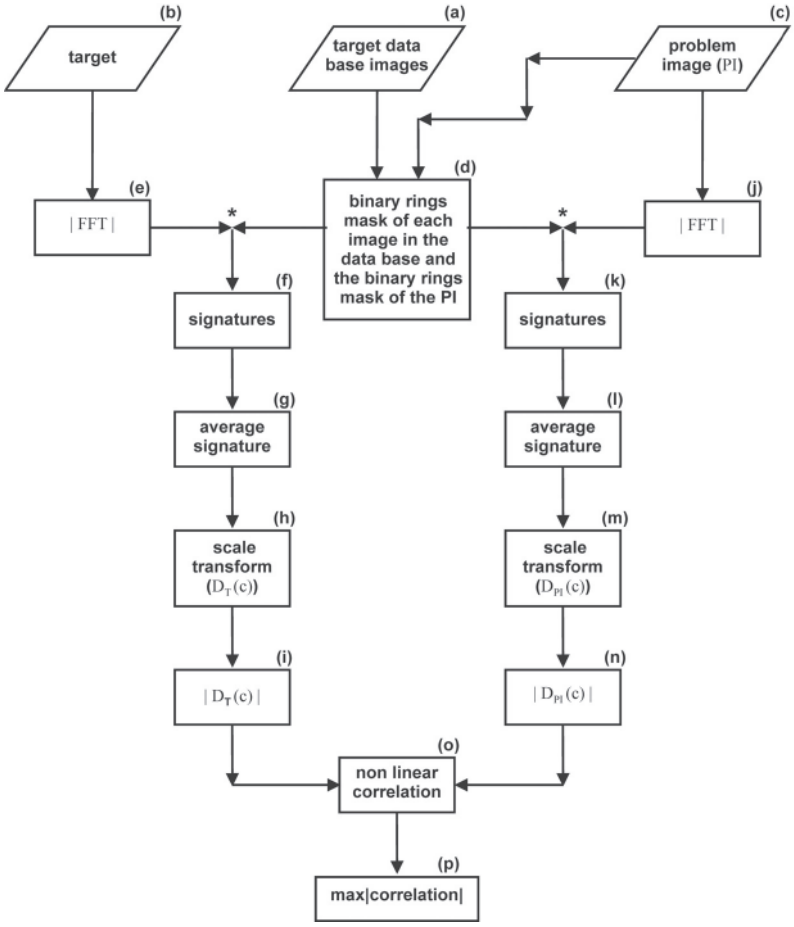


Fig. 11. Position, rotation and scale invariant non linear correlation digital system algorithm.

For example, Fig. 15(a) is chosen as the target. Then, its Fourier plane was made of training images using Eq. (5), shown in Fig. 15(b). Then, the corresponding binary rings mask is applied to it (Fig. 15(c)) to obtain the signature of the target (Fig. 15(d)). Analogously, the problem image is selected (Fig. 15(e)), the $|FFT(PI)|$ is calculated (Fig. 15(f)). The respective binary rings mask is applied (Fig. 15(g)) to construct the signature corresponding to the PI (Fig. 15(h)). Once it is obtained the signatures, the non linear correlation in Eq. (3) with $k = 0.1$ is applied (Fig. 15(i)). If the maximum value of the magnitude for the correlation of the problem image is significant, that is similar to the autocorrelation maximum value of the target, hence the PI contains the target, otherwise has a image different to the target (Fig. 15(j)).

In Fig. 16 is shown the box plot results when Fig. 14(a) is taken as target. As expected the correlations for the Fig. 14(a) are more separate from the rest, showing the digital system

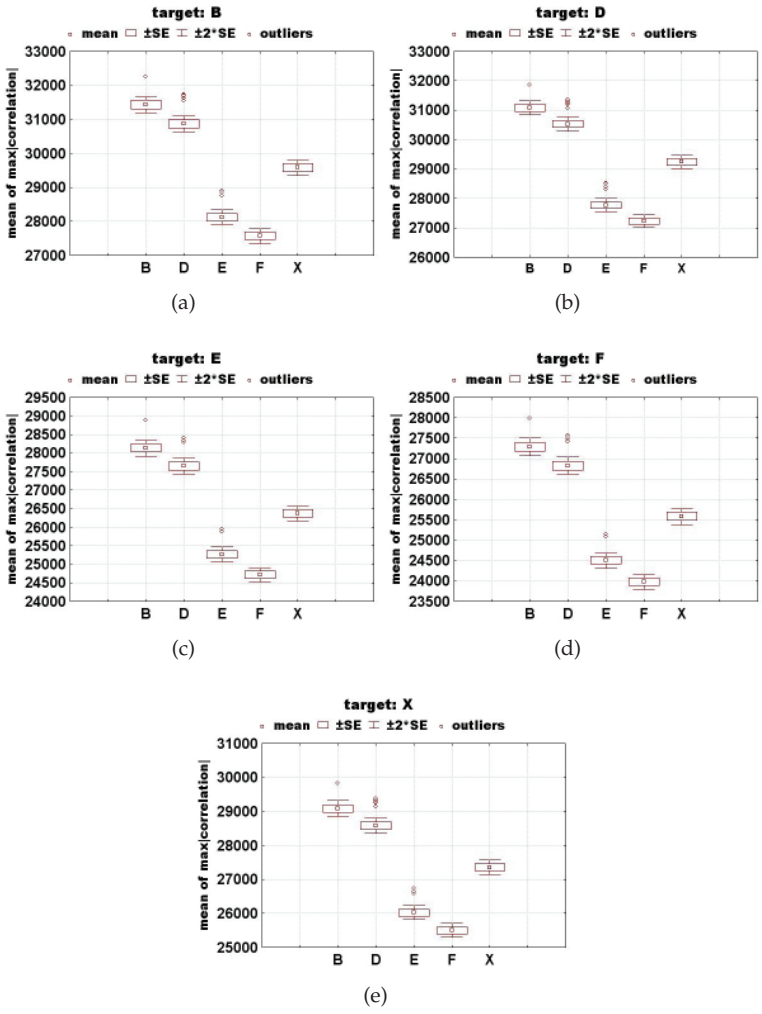


Fig. 12. Box plot analysis of images in Fig. 9.

has a good performance to discriminate between other images that do not correspond to this diatom.

In general if we compares these techniques with other methodologies, we can affirm that these algorithms are robust and they have low computational cost. The algorithms were programming in Matlab 7.1, in a MacBook Pro 3.1 with a Intel Core 2 Duo processor of 2.4 GHz, memory of 2 GB 667 MHz DDR2 SDRAM, L2 Cache of 4 MB and 800 MHz of Bus Speed. The time machine for the correlation per image were around 0.25 seconds. (Álvarez-Borrego & Solorza, 2010; Solorza & Álvarez-Borrego, 2010)

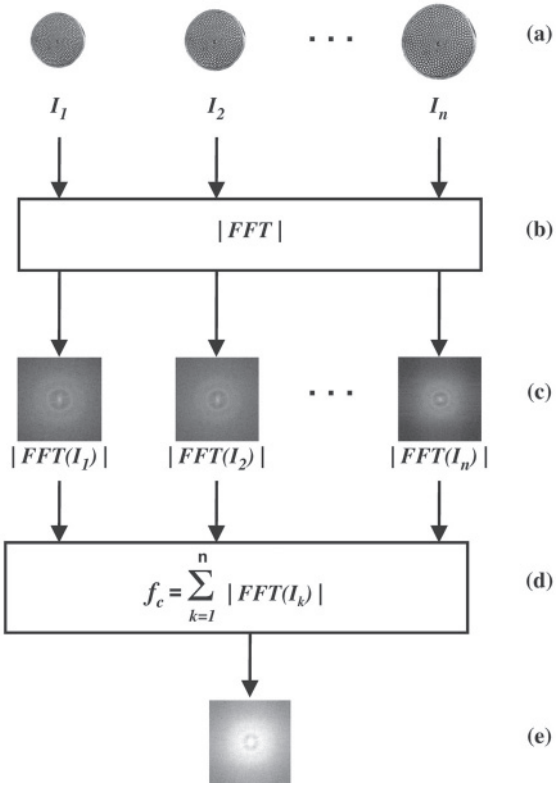


Fig. 13. Procedure to train a composite filter.

Fig. 14. Examples of diatoms.

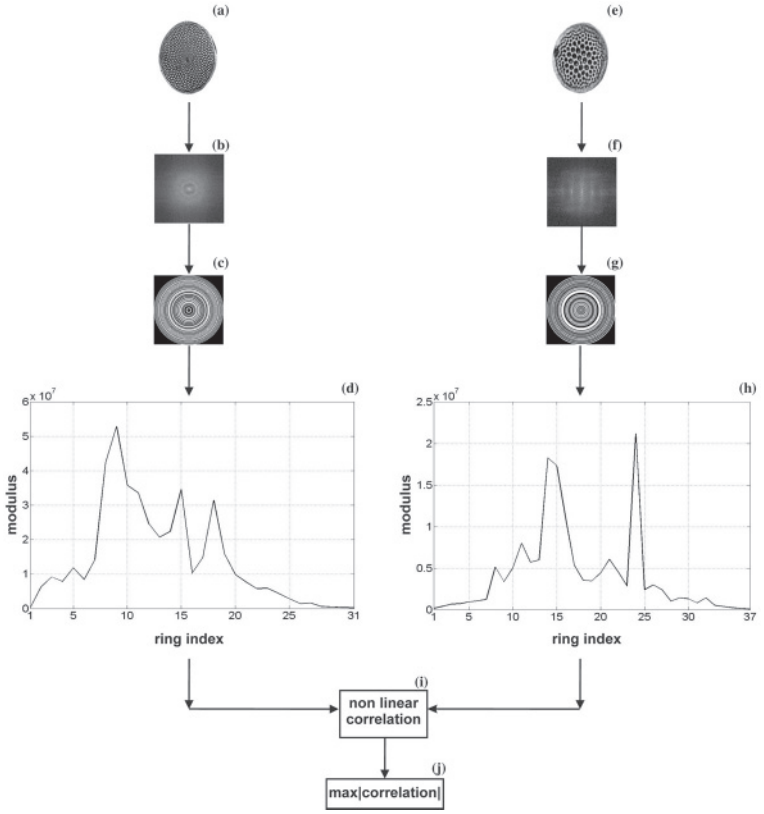


Fig. 15. Non linear correlation between signatures.

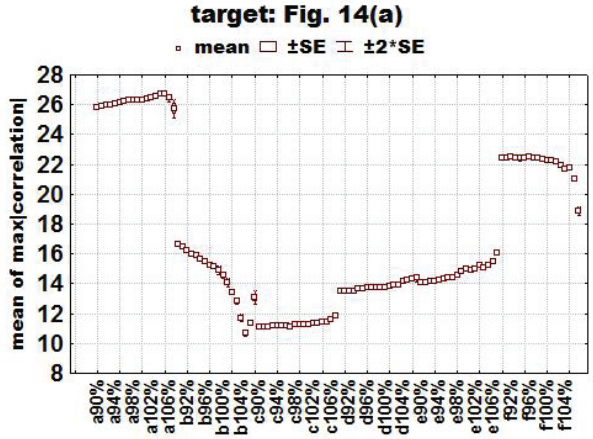


Fig. 16. Box plot example for diatom images in Fig. 14.

5. Conclusion

In this chapter was presented a simple and efficient non linear correlation digital systems invariant to position and rotation and other two digital systems that incorporate the scale invariance. Taking advantage of the translation property for the Fourier transform of a function the invariance to position was achieved. The relevant information of the images is captured in the average signatures via the adaptive mask, helping us on the discrimination between rotated objects. Moreover, the masks were constructed based on the given image; hence, the mask is adapted to problem avoiding in this form the information leak. Here is presented two forms to obtain the scale invariant. The first form uses the scale transform applied to the average signature of the images. The second is based on the Fourier plane to built a composite filters.

Because the signatures of the images are vectors, the computational time cost was reduced considerably compared to those systems that use bi-dimensional signatures (matrices). One of the advantages of these kinds of methodologies is that an entire process can be repeated over and over in the same way without mistakes.

In the particular examples presented in this chapter, the three digital systems present an excellent performance, a confidence level of 95.4% or greater. The digital system in Fig. 7 identified fingerprints images rotated ± 15 degrees. The algorithm in Fig. 11 classified contour letters which are translated, rotated and scaled. The composite filters with training images (Fig. 15) recognize the positional shifted, rotated and scaled diatoms images. The biggest advantage of these methodologies are the simplicity in their construction.

6. Acknowledgments

This work was partially supported by CONACyT with grant No. 102007 and PROMEP. The authors thank to Professors De Sena A. and Rocchesso D. by the Matlab routines to calculate the one-dimensional scale transform.

7. References

Alon, J., Athitsos, V., Yuan, Q. y Sclaroff, S. (2009). A Unified framework for gesture recognition and spatiotemporal gesture segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, Vol. 31, No. 9, 1685-1699, ISSN 0162-8828.

Álvarez-Borrego, J. & Castro-Longoria, E. (2003). Discrimination between Acartia(Copepoda: Calanoida) species using their diffraction pattern in a position, rotation invariant digital correlation. *Journal of Plankton Research*, Vol. 25, No. 2, 229-233, ISSN 0142-7873.

Álvarez-Borrego, J., Mouriño-Pérez, R.R., Cristóbal-Pérez, G. & Pech- Pacheco, J.L. (2002). Invariant recognition of polychromatic images of Vibrio cholerae 01. *Optical Engineering*, Vol. 41, No. 4, 827-833, ISSN 009 1-3286.

Álvarez-Borrego, J. & Solorza, S. (2010). Comparative analysis of several digital methods to recognize diatoms. *Hidrobiológica*, Vol. 20, No. 2, 158-170, ISSN 0188-8897.

Arandjelovic, R. & Zisserman, A. (2010). Efficient image retrieval for 3D structures. *Proceedings of the British Machine Vision Conference (BMVA)*, 30.1-30.11, ISBN 1-901725-40-5, Aberistwyth, UK, September 2010, BMVA Press, UK.

Barinova, O., Lempitsky, V., Kohli, P. (2010). On the detection of multiple object instances using Hough transforms. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2233-2240, ISBN 978-1-4244-6984-0, San Francisco, CA, USA, June 2010, IEEE Computer Society Press, USA.

Bueno-Ibarra, M.A., Chávez-Sánchez, M.C. & Álvarez-Borrego, J. (2011). K-law spectral signature correlation algorithm to identify white spot syndrome virus in shrimp tissues. *Aquaculture*, Vol. 318, 283-289, ISSN 0044-8486.

Coronel-Beltrán, A. & Álvarez-Borrego, J. (2010). Comparative analysis between different font types and letter styles using a nonlinear invariant digital correlation. *Journal of Modern Optics*, Vol. 57 No. 1, 1-7, ISSN 0950-0340.

De Sena, A. & Rocchesso, D. (2004). A study on using the Mellin transform for vowel recognition. *Proceedings of the International Conference on Digital Audio Effects*, 5-8, ISBN 88-901479-0-3, Naples, Italy, October 2004, DAFx' 04, Naples, Italy.

Díaz-Ramírez, V.H., Kober, V. & Álvarez-Borrego, J. (2006). Pattern recognition with an adaptive joint transform correlator. *Applied Optics*, Vol. 45, No. 23, 5929-5941, ISSN 1559-128X.

Fergus, R., Perona, P. & Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 264, ISBN 0-7695-1900-8, Madison, Wisconsin, USA, June 2003, IEEE Computer Society Press, USA.

Forero-Vargas, M., Cristóbal-Pérez, G. & Álvarez-Borrego, J. (2003). Automatic identification techniques of tuberculosis bacteria. *Proceedings of SPIE: Applications of Digital Image Processing XXVI*. 5203, 71-81, ISBN 0819450766, San Diego, CA, USA, August 2003, SPIE Press, USA.

González-Fraga, J.A., Kober, V. & Álvarez-Borrego, J. (2006). Adaptive SDF filters for pattern recognition. *Optical Engineering*, Vol. 45, No. 5, 057005, ISSN 009 1-3286.

Gonzalez, R.C. & Woods, R.E. (2002). *Digital image processing*, Prentice Hall, ISBN 0-201-18075-8, USA.

Guerrero-Moreno, R.E. & Álvarez-Borrego, J. (2009). Nonlinear composite filter performance. *Optical Engineering*, Vol. 48, No. 6, 06720, ISSN 009 1-3286.

Holub, A.D., Welling, M. & Perona, P. (2005). Combining generative models and Fisher kernels for object recognition. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 136-143, ISBN 0-7695-2334-X, Beijing, China, October 2005, IEEE Computer Society Press, USA.

Jain, A.K. & Feng, J. (2011) Latent fingerprint matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 33, No. 1, 88-100, ISSN 0162-8828.

Komarinski, P., Higgins, P.T., Higgins, K.M. & Fox, L.K. (2005). *Automated fingerprint identification system (AFIS)*, Elsevier Academic Press, ISBN 0-12-418351-4, San Diego, CA, USA.

Kong, H., Audibert, J. & Ponce, J. (2010a). Detecting abandoned objects with a moving camera. *IEEE Transactions on Image Processing*, Vol. 19, No. 8, 2201-2210, ISSN 1057-7149.

Kong, H., Audibert, J. & Ponce, H. (2010b). General road detection from a single image. *IEEE Transactions on Image Processing*, Vol. 19, No.8, 2211-2220, ISSN 1057-7149.

Lempitsky, V., Zisserman, A. (2010). Learning to count objects in images. *Advances in Neural Information Processing Systems 23*, Curran Associates Inc, ISBN 1615679111, NY, USA.

Lerma-Aragón, J. R. & Álvarez-Borrego, J. (2009a). Vectorial signatures for invariant recognition of position, rotation and scale pattern recognition. *Journal of Modern Optics*, Vol. 56, No. 14, 1598 -1606, ISSN 0950-0340.

Lerma-Aragón, J.R. & Álvarez-Borrego, J. (2009b). Character recognition basen on vectorial signatures. *e-Gnosis*, Vol. Concibe, No. 9, 1-7, ISSN 1665-5745.

Moses, K.R., Higgings, P., McCabe, M., Probhakar, S. & Swann, S. (2009). Automatic fingerprint identification systems (AFIS), In: *The fingerprint sourcebook*, International Association for Identification. National Institute of Justice, Washington, DC, USA.

Pech-Pacheco, J.L. & Álvarez-Borrego, J. (1998). Optical-digital system applied to the identification of five phytoplankton species. *Marine Biology*, Vol. 132, No. 3, 357-366, ISSN 0025-3162.

Ponce, J., Berg, T.L., Everingham, M., Forsyth, D.A., Hebert, M., Lazebnik, S., Marszalek, M., Schmid, C., Russell, B.C., Torralba, A., Williams, C.K.I., Zhang, J. & Zisserman, A. (2006). Dataset issues in object recognition, In: *Toward category-level object recognition (lectures notes in computer science/image processin, computer vision, pattern recognition, and graphics)*, Ponce, J., Hebert, M., Schmid, C. & Zisserman, A., 29-48, Springer, ISBN 3-540-68794-7, Berlin.

Solorza, S. & Álvarez-Borrego, J. (2010). Digital system of invariant correlation to position and rotation. *Optics Communications*, Vol. 283, No. 19, 3613-3630, ISSN 0030-4018.

Zavala-Hamz, V. & Álvarez-Borrego, J. (1997). Circular harmonic filters for the recognition of marine microorganisms. *Applied Optics*, Vol. 36, No. 2, 484-489, ISSN 1559-128X.

14

Vectorial Signatures for Pattern Recognition

Jesús Ramón Lerma-Aragón[1] and Josué Álvarez-Borrego[2]
[1]*Facultad de Ciencias, Universidad Autónoma de Baja California,*
[2]*CICESE, División de Física Aplicada, Departamento de Óptica*
México

1. Introduction

Due to the variety of shapes and sizes that present both living organisms and static objects, the necessity to look for automated systems of identification, both in industry and scientific research has arisen. During the 1960s, the scientific community in the field of optics has used the Fourier transform and other types of mathematical transformations for pattern recognition, taking advantage of their different properties; for example, invariance to position, rotation and scale.

Many invariant descriptors use the Fourier transform to extract invariant features. The Fourier transform has been a powerful tool for pattern recognition. One important property of the Fourier transform is that a shift in the time domain causes no change in the magnitude spectrum. This can be used to extract invariant features in pattern recognition.

In the last few years, they have been used as a tool in the digital processing of pattern recognition. Recently several books have been published (Pratt, 2007, Gonzalez et al., 2008 and 2009, Cheriet et al., 2007, and Obinata, 2007), that show us a general view, the progress and development of pattern recognition, as well as different tools for image pre-processing, extraction, selection and creation of features, classification methods, using different types of transforms, etc. However, some studies have focused on solving the problem of invariance to rotation and scale (Cohen, 1993, Casasent, 1976a, Pech et al, 2001 and Pech et al, 2003, Solorza and Álvarez-Borrego, 2010, Solorza and Álvarez-Borrego, 2011).

For practical implementation of the theory it should be considered, according to Casasent and Psaltis (1976a, 1976b, 1976c), a system that is invariant to scale through the manipulation of the Fourier transform directly changing the input function. We must also consider the practical realization of the Mellin transformation given by a logarithmic mapping of the input stage followed by a Fourier transform.

Schwartz (1994) found that there is strong evidence in many physiological and psychophysical visual systems (including human) use of such logarithmic mapping between the retina and visual cortex. The motion of changing a scaling to a shift by a logarithmic transformation of the coordinates occurs in many areas. In fact, the log-polar coordinate system described above seems to have a biological analogue. Since biological systems can shift objects to the centre of the field of view by movement of the eyes, it seems logical that a mapping that facilitates scale and rotation invariant recognition would be most useful. This is not to say that biological systems also compute Fourier transforms of these representations.

Other methods can take advantage of representation in which transformations are reduced to shifts.

The different techniques used today have difficulties, the main one being that the calculations involve a high computational complexity. Moreover, in this case, the complexity of the operation performed is directly proportional to the resolution of the images used. The computational cost associated inevitably increase the time required for comparison. Emerging applications of computer vision and pattern recognition require the development of new algorithms. The vectorial signatures techniques have an important role to play in this context, given their simplicity and low computational requirements.

In this chapter we will focus on examining the capabilities of vectorial signatures, as a method of recognizing objects, and we will consider inside this algorithm invariance to position, rotation and scale. It will be applied here to the recognition of alphabetical letters in Arial typeface, copepod species and some butterflies species (color images). However, this algorithm can be applied to any kind of object. The system was developed by taking advantage of the properties of the Fourier, Mellin and Scale transforms.

2. Mathematical description

In this section we present some basic concepts, such as the definition of Fourier, Mellin and Scale transforms, tools that are of great usefulness in audio and image processing, noise reduction signals (such as white noise) frequency analysis of any discrete signal, materials analysis and statistical synthesis by inverse Fourier transform, and so on.

2.1 Fourier transform

The Fourier transform $F(u)$ of a continuous function $f(x)$ of a single variable is defined

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux} dx \tag{1}$$

Where $j = \sqrt{-1}$. Consequently, the inverse Fourier transform (IFT) of $F(u)$ results in the original function, described by

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du \tag{2}$$

2.1.1 The Fourier-Mellin transform

Mellin transform is especially useful for the scale invariant (Bracewell, 1978). This has been applied to the spatially varying image restoration and in the analysis of networks that vary with time, among others.

The two dimensional Mellin transform $M(ju,jv)$ of a function $f(x,y)$ along the imaginary axis is defined by Casasent and Psaltis (1976a)

$$M(ju, jv) = \int_{0}^{\infty}\int_{0}^{\infty} f(x,y) x^{-ju-1} y^{-jv-1} dxdy \tag{3}$$

2.1.2 The scale transform

Cohen (1993) introduced the "scale transform." This transform is said to be scale invariant, thus meaning that the signals differing just by a scale transformation (compression or expansion with energy preservation) have the same transform magnitude distribution. Cohen showed that the scale transform is a restriction of the Mellin transform on the vertical line $p = -jc + 1/2$, with $c \in R$.

We choose the scale transform, which is a restriction of the Fourier-Mellin transform (Casasent, 1976c). The main property of the scale transform is the scale invariant property.

If we call c the scale variable, then in two dimensions the scale transform and its inverse is given by Cristobal & Cohen (1996).

$$D(c_x, c_y) = \frac{1}{\sqrt{2\pi}} \int_0^\infty \int_0^\infty f(x,y) e^{\left(-jc_x Inx - jc_y Iny\right)} \frac{dxdy}{\sqrt{xy}}, \tag{4}$$

$$f(x,y) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} D(c_x, c_y) e^{\left(jc_x Inx + jc_y Iny\right)} \frac{dc_x dc_y}{\sqrt{xy}}. \tag{5}$$

3. Implementation

In this section we will use two-dimensional scale transformations now written in polar coordinates $D(c_r,c_\theta)$, because this transform is invariant to changes in size and rotation

$$D(c_r, c_\theta) = \frac{1}{\sqrt{2\pi}} \int_0^\infty \int_0^{2\pi} f(r,\theta) r^{\left(-jc_r - \frac{1}{2}\right)} e^{(-jc_\theta \theta)} dr d\theta, \tag{6}$$

and taking the logarithm of the radial coordinate, that is $\lambda = Inr$, we get

$$D(c_\lambda, c_\theta) = \frac{1}{\sqrt{2\pi}} \int_0^\infty \int_0^{2\pi} f(\lambda,\theta) r^{\left(-jc_r - \frac{1}{2}\right)} e^{\frac{\lambda}{2}} e^{(\lambda c_\theta + \theta c_\theta)} d\lambda d\theta. \tag{7}$$

Applying the transformation described above, we built a new algorithm. Fig. 1 shows the block diagram of the methodology used. The target (I^i) to be recognized is denoted by the function *f(x,y)* (step 1) and the modulus of its Fourier transform *(F)* is obtained (step 2). A parabolic filter to the modulus of *F* is applied (step 3). In this way, low frequencies are attenuated and high frequencies are enhanced in proportion to $(w_x)^2$, $(w_y)^2$ (Pech-Pacheco et al., 2003), where w_x and w_y are the angular frequencies. The scale factor $\sqrt{r}$, (step 4) is applied to the results of step 3 (r is the radial spatial frequency). This process is what differentiates the scale transform from the Mellin transform. After these steps, we mapped the Cartesian coordinates to polar-logarithmic coordinates (step 5), to obtain invariance to rotation. In this step we introduced a bilinear interpolation of the first data of coordinate conversion. This is done to avoid the aliasing due to the log-polar sampling. A subimage denoted by M(λ,θ) (Lerma-Aragón & Álvarez-Borrego 2008) is chosen from step 5, which contains more information of the target and a bilinear interpolation is done again in order to resize the subimage data to its original size (step 6). From the subimage, we obtain two 1-D

vectors by projecting M(λ,θ) onto the x and y axes (steps 7 and 8). In other words, we compute the marginals, according to the equations

$$M(\lambda_m) = \sum_{k=1}^{n} M(\lambda_m, \theta_k), \tag{8}$$

$$M(\theta_n) = \sum_{k=1}^{m} M(\lambda_k, \theta_n). \tag{9}$$

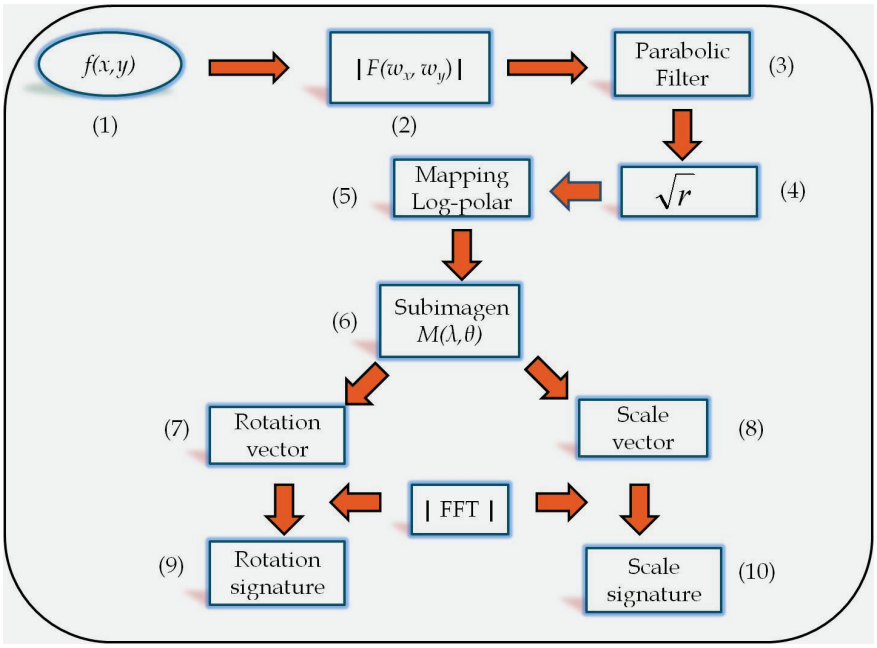


Fig. 1. Block diagram of used procedure

A modulus of the Fourier transform is calculated in order to obtain the rotation and scale signatures (equations 10 and 11), which will be the two unidimensional vectors for the target (steps 9 and 10).

$$V_1(w_\lambda) = \left| F\left[M(\lambda_m) \right] \right|, \tag{10}$$

$$V_2(w_\theta) = \left| F\left[M(\theta_n) \right] \right|, \tag{11}$$

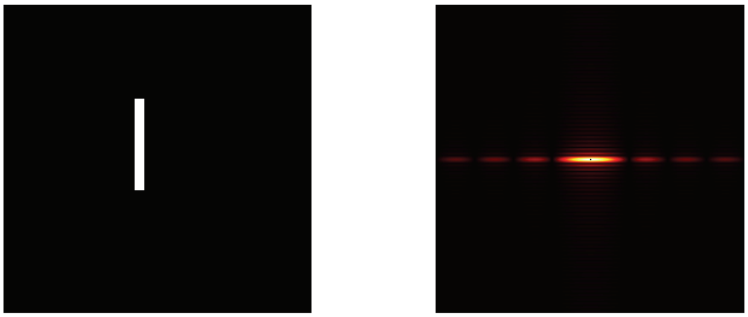
To evaluate this algorithm, the target image (I^i) and an image problem (I^j) (I^j is denoted by the function $g(x,y)$ either rotated or scaled) are selected, to which we apply the procedure described above. To determine the similarity between the images I^i and I^j in the database, using the Euclidean distance (E_d) is calculated among their average firms using the following equation

$$E_d = \sqrt{\sum \left\{ \left[V_{1f}(w_\lambda) - V_{1g}(w_\lambda) \right]^2 + \left[V_{2f}(w_\theta) - V_{2g}(w_\theta) \right]^2 \right\}}. \quad (12)$$

Thus, the signatures of all the *j* images of the database can be compared with any image *i* to be recognized.

3.1 Application of the algorithm

To apply the algorithm based on the theory described above, is considered primarily a target image, in this particular case, we consider a simple image (rectangle) of size 256 X 256 pixels and black background, as shown in Fig. 2(a), which is calculated the magnitude of the Fourier transform, which can be seen in Fig. 2(b). We used Matlab colormap command to change to a colour figure and of this way clearly visualize each step of the procedures. When these values are scaled linearly for display, the brightest pixels will dominate the display, at the expense of lower (and just as important) values of the spectrum. If, instead of displaying the values in this manner, we apply logarithmic transformations (Gonzalez, 2009) to the spectrum values, then the range of values of the result is a more manageable number. Fig. 3 shows the result of scaling this new range linearly and displaying the spectrum. The wealth of detail visible in this image compared to a straight display of the spectrum is evident from these pictures.



(a) (b)

Fig. 2. (a) Simple image of a rectangle. (b) The corresponding centered spectrum

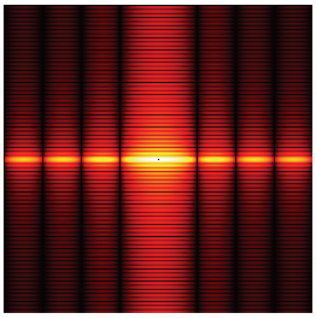


Fig. 3. Result showing increased detail after a log transformation

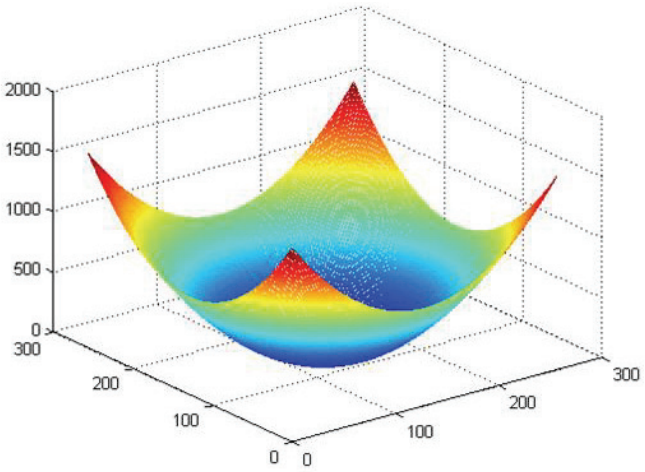
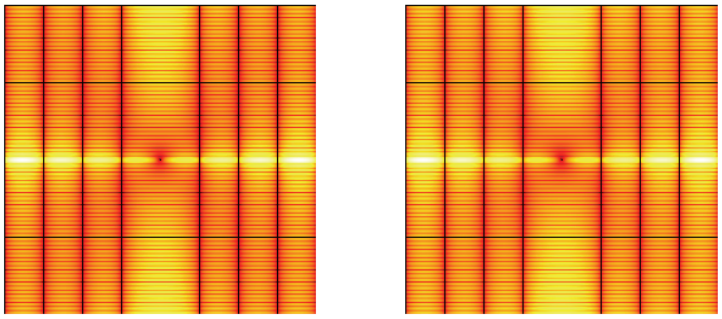


Fig. 4. Parabolic filter



(a) (b)

Fig. 5. (a) Modulus with parabolic-effect. (b) Image with scale factor

A parabolic filter (Fig. 4) to the modulus of F is applied, in this way, low frequencies are attenuated and high frequencies are enhanced as shown in Fig. 5(a). The scale factor $\sqrt{r}$, is applied to the previous results (Fig. 5b). A subimage is chosen from Fig. 6(a) corresponding to the zone that contains the major quantity of information. In the Fig. 6(b) results are observed.

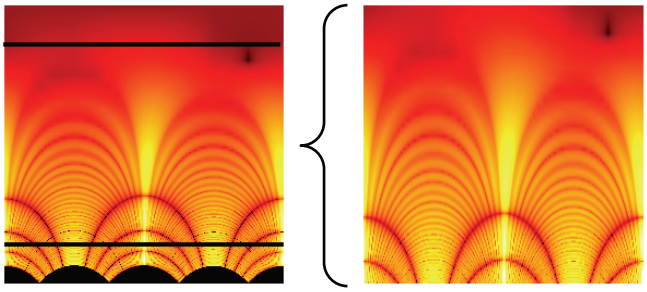
From the subimage $M(\lambda,\theta)$ seen in Fig. 6(b), we compute the marginal functions, using the following equations

$$g(x) = \sum_{\theta} M(\lambda,\theta), \tag{13}$$

$$h(x) = \sum_{\lambda} M(\lambda,\theta)., \tag{14}$$

In the Fig. 7(a) the subimage (Fig. 6b) appears, in three dimensions, and in 7(b) and 7(c) the marginal functions can be observed, with which scale and rotation vectors are obtained.

Finally, Fig. 8 shows the modulus of the Fourier transform of the vectors showed in the Fig. 7(b) and 7(c) respectively, which defines the vectorial signatures of the image, and they are used for comparison with respect to another image.



(a) (b)

Fig. 6. (a) Image 5(b) in polar-logarithmic coordinates. (b) Selected subimage

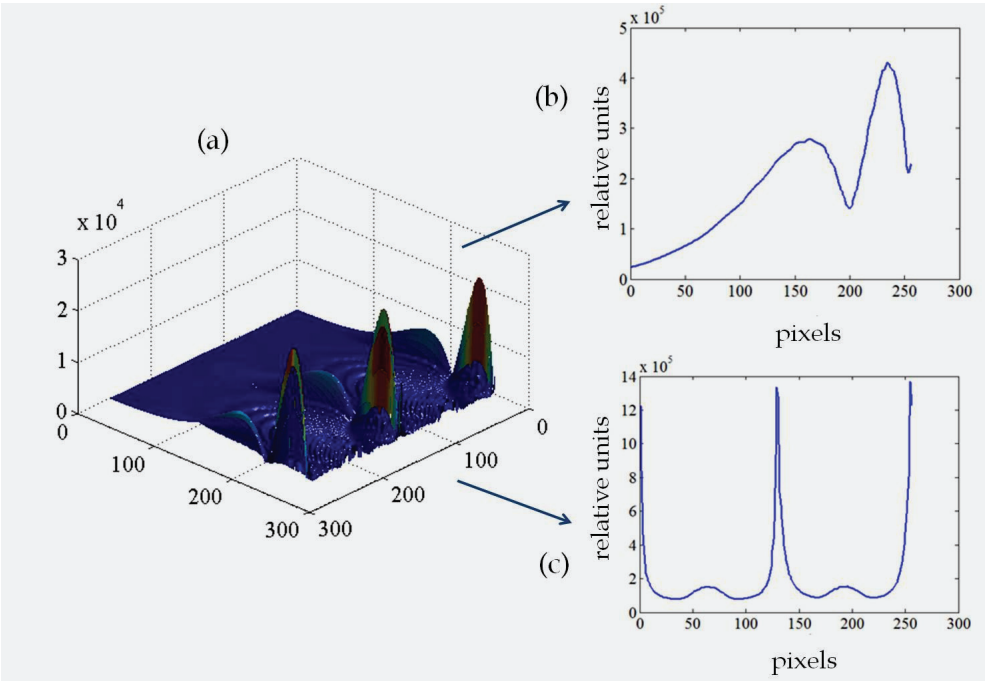


Fig. 7. (a) Tridimensional image of Fig. 6(b). (b) Vector of the summation on the rotation axis. (c) Vector of the summation on the scale axis

4. Computer simulations in grayscale images

To evaluate the performance of the vectorial signatures, images of different types were used as input images.

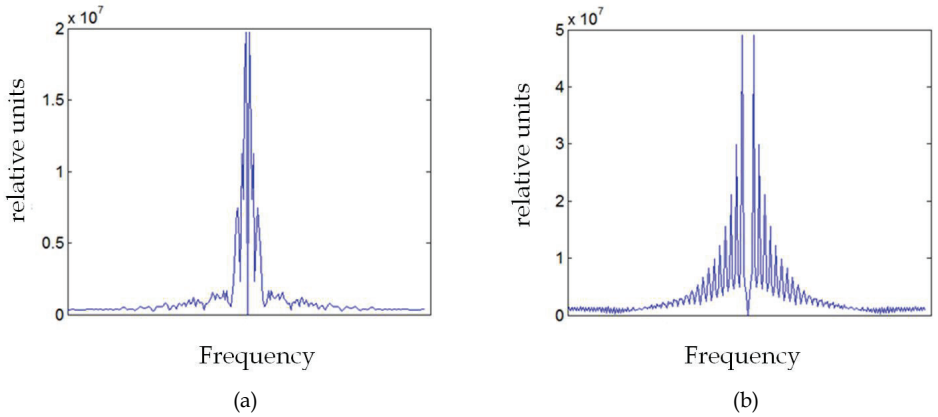


Fig. 8. (a) Scale signature vector. (b) Rotation signature vector

4.1 Simulation using letters

The first case analyzed corresponds to images of the different letters of the alphabet. Each letter is an image of 256 X 256 pixels of black background with a centered white Arial letter size 72. The methodology described above was applied to these letters. The behavior of the Euclidean distance (E_d) for the target E versus itself with respect to rotation from 0 to 359º with variations of one degree was studied. Because of the letter distortions, when is rotated, due to the square pixels, the curve representing the E_d is cyclic and with symmetry of 180º (Fig. 9).

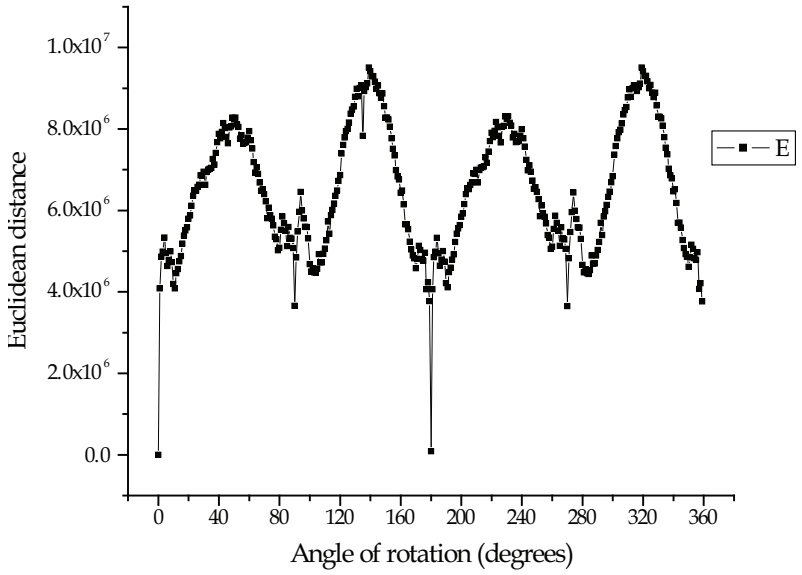


Fig. 9. Behavior of the Euclidean distance where the target and the input scene is the letter E

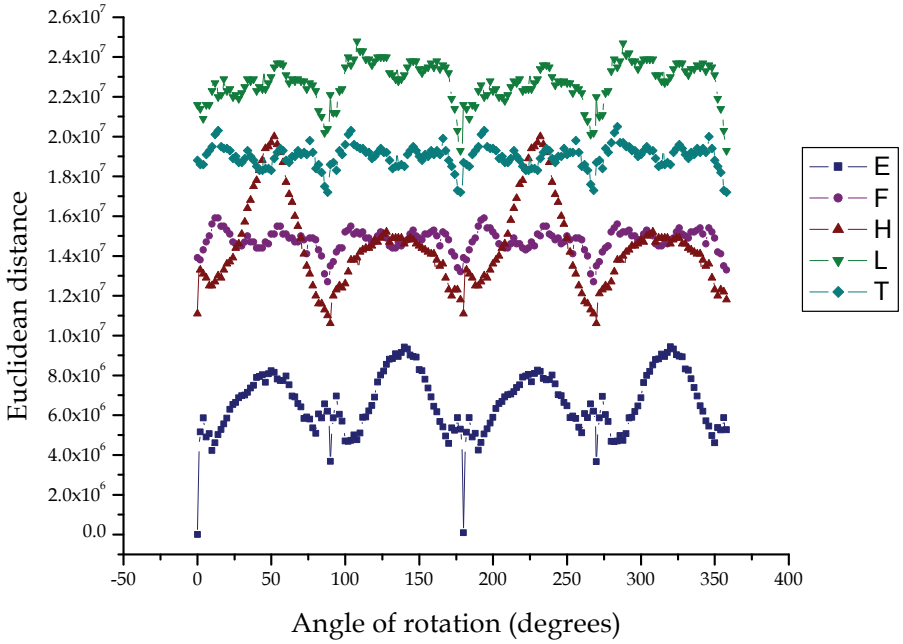


Fig. 10. Euclidean distance of the letter E compared with other letters

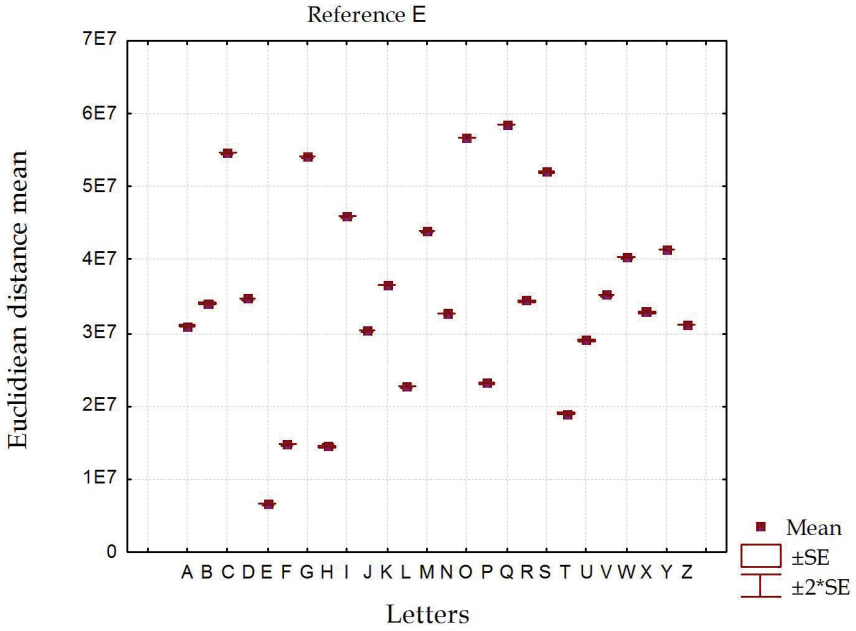


Fig. 11. Statistical distance behavior

To evaluate the distinguish ability between the 26 letters of the alphabet; each one was rotated 360 degrees, with a variation of one degree. Simulation was performed to determine the difference between them, using the 9360 images (26x360). In Figure 10 shows the behavior of the letters that have greater similarity with the reference (E); the result shows a very clear separation between the values of the Euclidean distance of the letter E compared with the rest of the letters, which allows being able to identify it. When an image is similar to another one, the E_d has a minimum value. Statistic was performed and the mean value ± 2SE (two standard error) was calculated. This algorithm has at least a 95.4% level of confidence for this case (Fig. 11 shows the comparison with all the letters).

To analyze the behavior to changes in scale, changes were tested by 90% to 110% with respect to the target image, with increments of 0.5%. Fig. 12. shows the results obtained by reference to the letter E and compared again with all the letters of the alphabet.

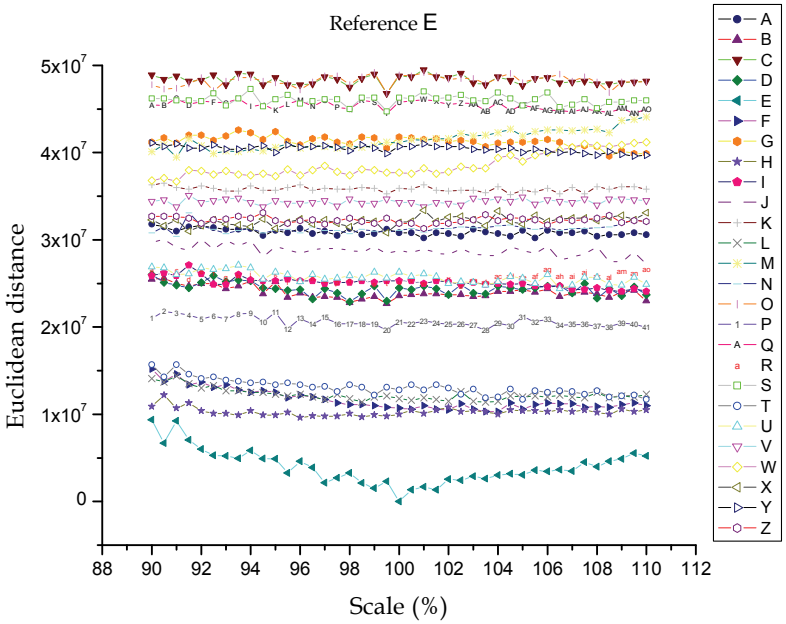


Fig. 12. Changes of distance versus scale

4.2 Simulation using copepod species

Copepods are the dominant group in marine zooplankton; they constitute at least 70% of the planktonic fauna. This group of organisms has a great diversity. There are about 11,500 species of copepods described by Humes (1994) and the number is steadily increasing with the description of new species, such as those from anchialine caves (Fosshagen, 1991), hydrothermal vents (Humes, 1991), and those previously reported under the name of another species (Bradford, 1976; Soh & Suh, 2000).

Copepods are of prime importance in marine ecosystems. The majority of copepods feed on phytoplankton, forming a direct link between primary production and commercially

important fish, such as sardine, herring, and pilchard. Copepods are also the main food source for a great variety of invertebrates. Studies on copepod abundance and species composition are particularly relevant, because most larvae of commercial fish feed on copepods. Hence, changes in the abundance of these plankters from year to year, may determine interannual population fluctuations of the commercially exploited fish stocks in a particular region.

For this study, adult stages of different copepod species were separated from several plankton samples. The specimens were observed with an optical microscope and their images were digitally captured using a charge coupled device camera (CCD). We used 14 images from seven different species of copepods each with male and female samples (Fig. 13). Since the background noise in all the images is mostly repetitive, the images where cleaned using some methods described en Guerrero & Álvarez-Borrego (2009). Fig. 14 shows the original and its pre-processed image for a female specimen *Rhincalanus nasutus.*

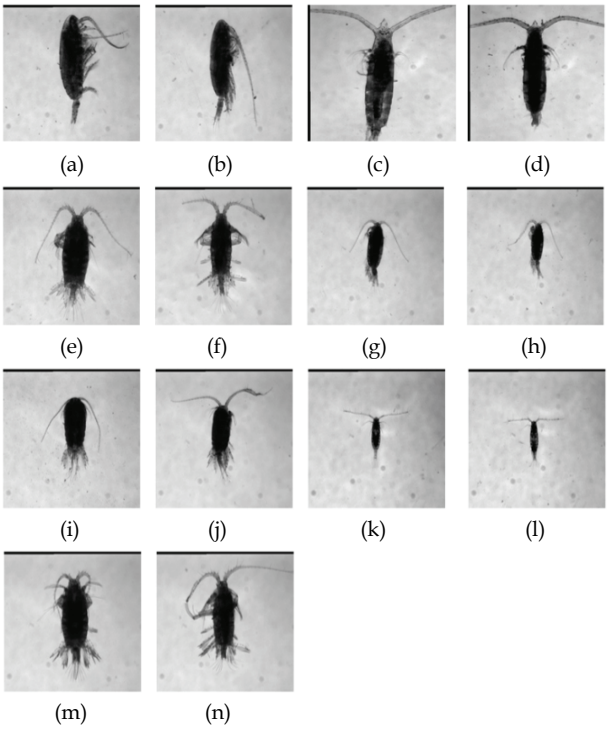
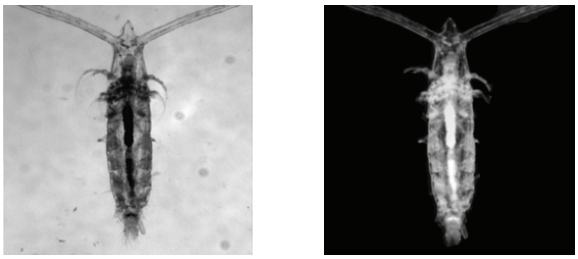


Fig. 13. Species of copepod used in the study: (a) *Calanus pacificus* female; (b) *C. pacificus* male; (c) *Rhincalanus nasutus* female; (d) *R. nasutus* male; (e) *Centropages furcatus* female; (f) *C. furcatus* male; (g) *Pleuromamma gracilis* female; (h) *P. gracilis* male; (i) *Temora discaudata* female; (j) *T. discaudata* male; (k) *Acartia tonsa* female; (l) *A. tonsa* male; (m) *Centropages hamatus* female; (n) *C. hamatus* male

The image of one of the calanoides *Calanus pacificus* female Fig. 13(a) was used as a target to discriminate between copepod species and sex from the rest of the organisms, the image size

is 256 X 256 pixels of black background with a centered copepod. The image was rotated 360º in increments of 1º to recognize a target in an input scene; signatures were compared using the Euclidean distance (E_d) between the target and the input image and, because there is no variation in the size of copepod adults, we did not analyze the changes of scale.

In order to see if this methodology has a good performance for recognizing the target image when compared with other images that don't correspond to the copepod chosen as reference, numerical calculations were performed for the entire set of 14 images showed in Fig. 13.



(a) (b)

Fig. 14. (a) Original image. (b) Cleaned images

Fig. 15. shows the performance of E_d for the fourteen pictures rotated 360 degrees. The Fig. 13(a) was used as a target, corresponding to a female *Calanus pacificus*. We can see a very good separation of the E_d for the target with respect to the calculation of the E_d for the non-target images. As can be seen in Fig. 15, the curve of the copepod (Fig. 13a) represents an E_d with minimum values, which distinguish it from the rest of the copepods. Statistic was realized and the mean value ±2 SE was calculated. We can see this algorithm has at least a 95.4% level of confidence for this case (Fig. 16).

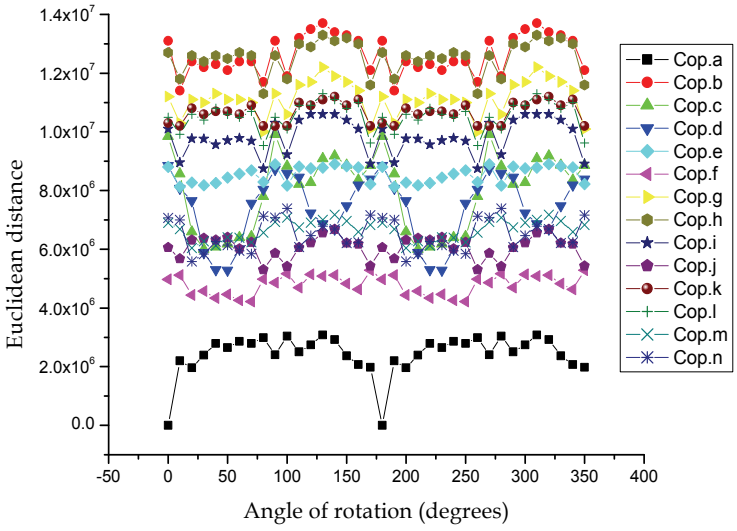


Fig. 15. Change the angle of rotation against E_d

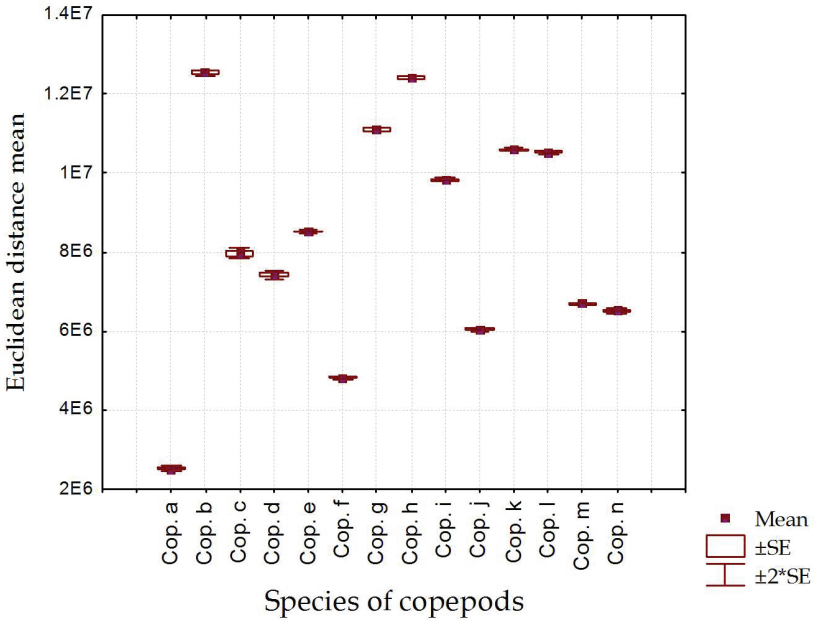


Fig. 16. Statistical behavior of the E_d, for a *Calanus pacificus* female specimen

4.3 Average signature

Like a second set of test images 30 females and 30 males from each of the 7 different species of copepods were used, thus forming a database of 420 images. To have an identification system that contains information from several copepods, vector composed signatures were designed, which is derived from adding the signatures of 10 different copepods and to get its average, which is now the reference vectorial signature (Fig. 17).

In Fig. 18 is shown the application of this method, we can see an excellent separation of the Euclidean distance mean, which corresponds to Centropages furcatus female (Group e), used like target, with respect to the calculation of the E_d from the rest of the groups. Statistic was realized and the mean value ± 2SE was calculated. We can identify the species and sex of the target with the confidence level was about 95.4%.

4.4 Comparison with other algorithms

To evaluate the performance of the algorithm, we compare it with respect to that published by Álvarez-Borrego and Castro-Longoria, considering that both are similar in application to pattern recognition and methodology based on the properties of the Fourier transform, in the second method is pattern matching using the correlation.

Comparing the results of the two algorithms applied to the same sets of images, we observed that in the second is not possible to distinguish the selected image as reference, with respect to the others. Regarding the computational cost, the processing time of our procedure is approximately 20% lower.

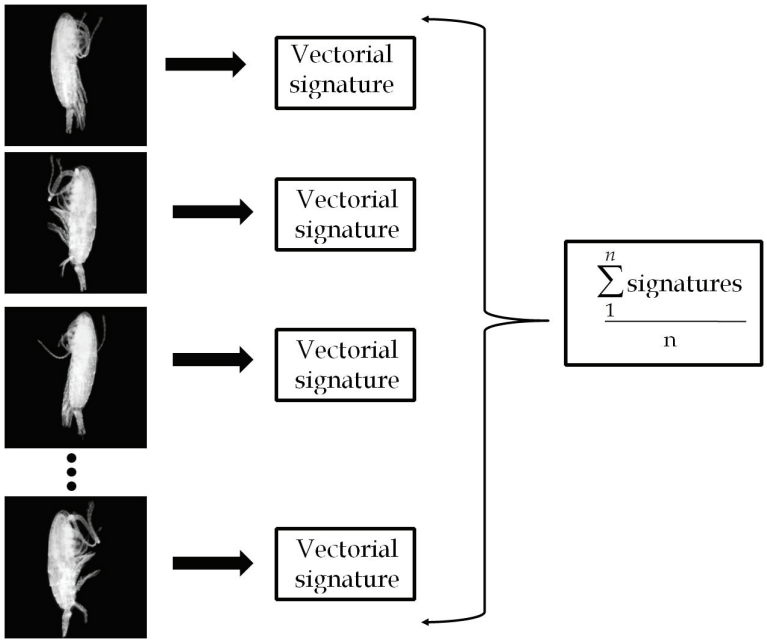


Fig. 17. Average vectorial signature

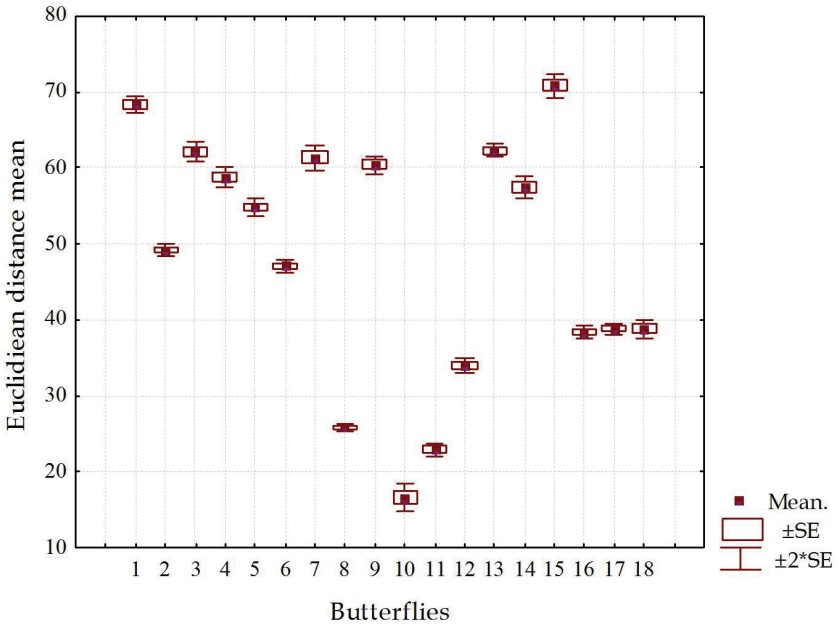


Fig. 18. Statistical behavior of the Euclidean distance, in the case of vectorial signatures composite of 10 images of Centropages furcatus female (Group e)

5. Computer simulations in color images

One of the most important characteristics, which account for visual pattern recognition is the color, and has been used in a wide range of applications in different areas of knowledge, such as evaluation and identification of textiles, flowers, microscopic images, surface corrosion, species recognition, and so on. In our daily life, our vision and actions are influenced by a variety of shapes and colors. The introduction of color increases the amount of information in pattern recognition, so that discrimination can be improved considerably.

Color is a perceived phenomenon and not a physical dimension like length or temperature. A suitable form of representation must be found for storing, displaying, and processing color images. This representation must be well suited to the mathematical demands of a color image processing algorithm, to the technical conditions of a camera, printer, or monitor, and to human color perception as well. These various demands cannot be met equally well simultaneously. For this reason, different representations are used in color image processing according to the goal. For these use different models that are known as color space (Gonzalez & Woods, 2008, Kosch & Abidi, 2008, Westland & Ripamonti, 2004) were used.

One of the most used color space is RGB, this model is based on the additive mixture of the three primary colors: red (R), green (G) and blue (B). When dealing with color images, there are several concepts that are inherently quite different. For example, if we treat the RGB signals at a pixel as a three-dimensional vector, a color image becomes a vector field, while a monochromatic image is a scalar field.

5.1 Methodology

Fig. 19. shows the block diagram of the methodology applied to RGB color spaces: in the first step, is select the color image to be used like a target (I^i) which is denoted by $f_i(x, y)$, where i = R,G,B, then each image is decomposed into its respective RGB channels [$f_R(x, y)$, $f_G(x, y)$, $f_B(x, y)$]. To have an identification system that contains information from the components $f_i(x,y)$, a vector-composed signatures were obtained, which is now the average vectorial signature $V_{1f}(w_\lambda)$ and $V_{2f}(w_\theta)$, for the test image (I^j) represented by $g_j(x, y)$ is performed the same process where were obtained $V_{1g}(w_\lambda)$ and $V_{2g}(w_\theta)$. To determine the similarity between the two images I^i and I^j, the value of the E_d is calculated.

5.2 Simulation for butterflies

To test this algorithm a set of images shown in Fig. 20 was used. With different types of butterflies, each image has a size of 256 x 256 pixels. In addition it is estimated that the number of species of butterflies in the world varies between 15.000 and 20.000, which according to the North American Butterfly Association (NABA), about 2000 are located in México.

In Fig. 21 we can see the statistical behavior of the Euclidean distance for the case of rotation of the image 360 degrees in one degree increments, using as reference the butterfly (10), which corresponds to a specimen *Archaeoprepona amphimachus amphiktion*, which at first glance is very similar to others butterflies, for example with butterfly (11) belongs to same genus but different species. As expected, the minimum value of the distance corresponds to the target. We can see that this algorithm has a very good performance in to recognize the target when is compared with the others butterflies. Numerical simulations were performed

considering as a reference each of the 18 butterflies and in all cases could be distinguished from the rest.

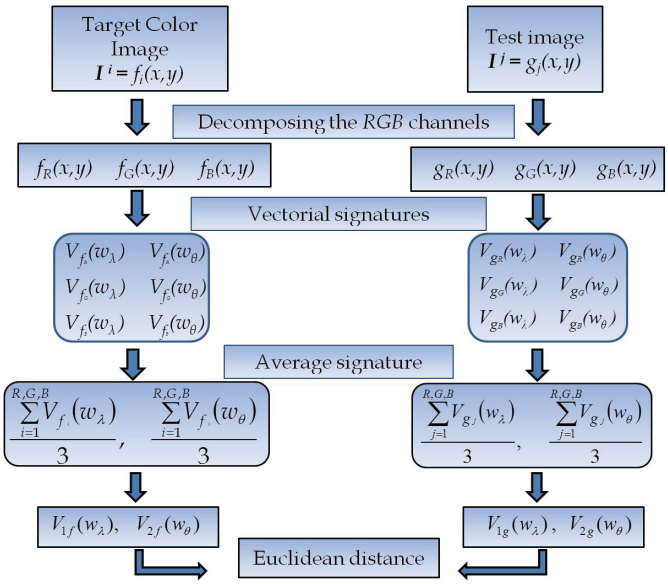


Fig. 19. Block diagram of used procedure

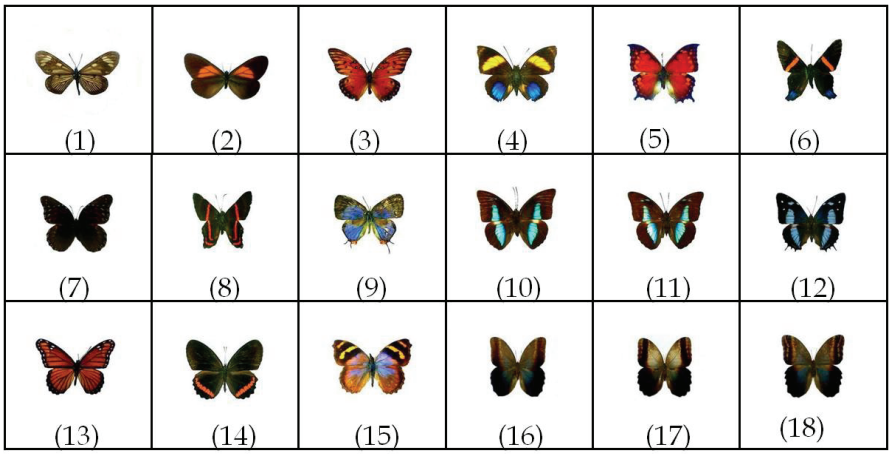


Fig. 20. Butterflies used. (1) Actinote guatemalena guerrerensis. (2) Actinote stratonice oaxaca. (3) Agraulis vanillae incarnata. (4) Agrias amydon oaxacata. (5) Anae aidea. (6) Ancyluris inca mora. (7) Anetia thirza. (8) Ansyluris jurgennseni. (9) Arawacus sito. (10) Archaeoprepona amphimachus amphiktion. (11) Archaeoprepona demophon centralis. (12) Baeotus baeotus. (13) Basilarchia archippus. (14) Biblis hyperia aganissa. (15) Bolboneura Sylphis sylphis. (16) Caligo eurylochus sulanos. (17) Caligo memnon. (18) Caligo oileus scamander

To analyze the invariance to scale changes were made from 70% to 130%, with variations of one percent each of the test images. The analysis of the statistical behavior of the distance is shown in Fig. 22. for the case where the butterfly (3) is used like target. The results indicate that there is separation of the reference image with the rest of the butterflies, which can be identified without difficulty, regardless of changes of scale.

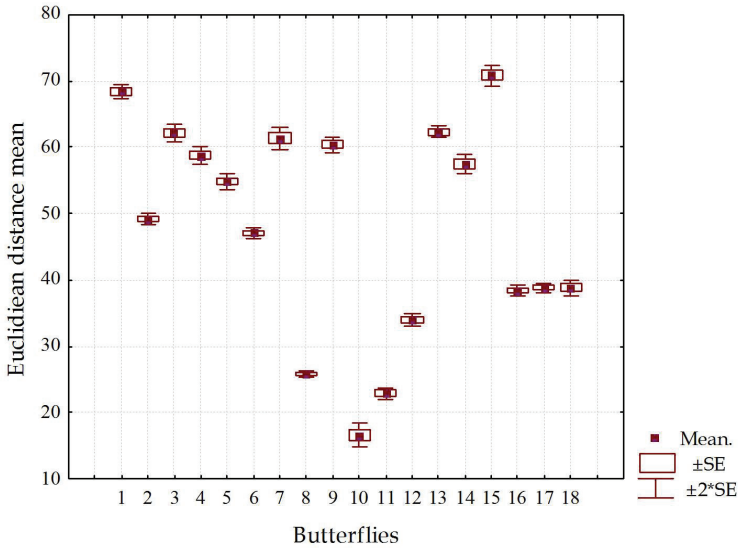


Fig. 21. Statistical behavior of the E_d, for an *Archaeoprepona amphimachus amphiktion* specimen

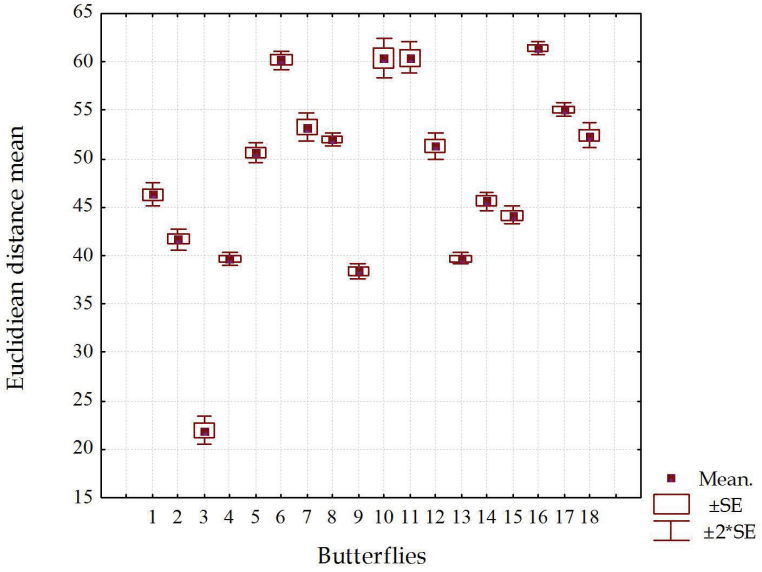


Fig. 22. Statistical behavior of the E_d, for a *Agraulis vanillae incarnata* specimen

6. Conclusion

In this chapter it was developed a system for pattern recognition with the utilization of vectorial signatures, based on the well-known relation between scale and Fourier transform, and has been performed to be practical and accurate. To test the built model, simulations were made and the results were analyzed. Conducting various numerical tests with letters of the alphabet, with variations in the scale and rotation, the results obtained from a comparison of the Euclidean distances show that it is possible for identification and discrimination with a high confidence level (at least 95.4%). The system was applied to the recognition of copepod species and sex. With vectorial signatures it is possible to identify, with a confidence level of 95.4%, the species and sex of various species of copepods. A modification of the algorithm in color image recognition, using average vector signature from the information contained in the RGB channels was done. For this case images of real butterflies were used, finding that the system can identify the species you choose as a reference, when compared with images of different butterflies, for any angle of rotation that contains, as well as changes in the scale of 70% to 130%.

This work contributes to increase the potential recognition systems for use in applications requiring assessment and interpretation of data, applications that were traditionally performed by human vision or specialists trained technicians and are still far from automation.

7. Acknowledgment

This document is based on work partially supported by UABC and CONACYT under Project grants 102007 and 169174.

8. References

Álvarez-Borrego J. & Castro-Longoria E., (2003). Discrimination between Acartia (Copepoda: Calanoida) species using their diffraction pattern in a position, rotation invariant digital correlation, *Journal of Plankton Research*, Vol. 25, No. 2, (February, 2003), pp 229-233, ISSN: 0142-7873.

Bracewell, R. N. (1999). *The Fourier transform and its applications*, (3th ed), McGraw-Hill, ISBN-10: 0073039381, New York.

Bradford, J. M. (1976). Partial revision of the Acartia subgenus Acartiura (Copepoda: Calanoida: Acartidae), *New Zealand Journal of Marine and Freshwater Research*., Vol. 10, No. 1, (March, 1976), pp. 159-202, ISSN: 0028-8330.

Casasent, D. & Psaltis, D. (1976a). Scale invariant optical correlation using Mellin transforms. *Optics Communication,* Vol. 17, No. 1, (April, 1976), pp. 59-63, ISSN: 0030-4018.

Casasent, D. &Psaltis, D. (1976b). Scale invariant optical transforms. *Optical Engineering,* Vol. 15, (May-June, 1976), pp. 258-261, ISSN: 0091-3286.

Casasent, D. & Psaltis, D. (1976c). Position, rotation, and scale invariant optical correlation. *Applied Optics* Vol. 15, No. 7, (July, 1976), pp. 1795-1799, ISSN: 0003-6935.

Cheriet, M., Kharma N., Liu C., Suen, C. (2007). *Character recognition systems. A guide for students and practioners*, John Wiley & Sons, Inc., ISBN: 978-0-471-41570-1, Hoboken, New Jersey.

Cohen, L. The Scale Representation, *IEEE Transactions on signal processing,* Vol. 41, No. 12, (December 1993), pp. 3275-3292. ISSN: 1053-587X

Cohen, L.(1993). The scale representation, *IEEE Transactions on Signal Processing,* Vol. 41, No. 12, (December 1993), pp.3275-3291, ISSN: 1053-587X.

Cristobal, G. & Cohen, L. (1996). Scale in image, *SPIE Proceedings*. Vol. 2846, pp. 251-261, Denver, CO, USA, August 1996.

Fosshagen, A. & Iliffe T, (1991), A new genus of calanoid copepod from an anchialine caves in Belize, *Bull. Plankton Soc. Japan*, Spec.Vol, pp. 339-346, ISSN: 0387-8961.

Gonzalez, R. & Woods, R. (2008). *Digital Image Processing*, (3rd Edition), Pearson Prentice Hall, ISBN: 978-0-13-168728-8, Upper Saddle River. New Jersey.

Gonzalez, R. C. & Woods, R.E. Eddins, S. L. (2009). *Digital Image Processing Using MATLAB®*, (2th ed), Gatesmark Publishing, ISBN: 978-0-9820854-0-0, USA.

Guerrero, R.E. & Álvarez-Borrego, J. (2009). Nonlinear composite filter performance, *Optical Engineering*. Vol. 48, No. 6, pp. 067201 1-11, ISSN: 0091-3286.

Humes, A. G. (1991). Zoogeography of copepods at hydrothermal vents in the eastern Pacific Ocean, *Bull. Plankton Soc. Japan*, pp. 383-389, ISSN: 0387-8961.

Humes, A. G. (1994). How many copepods?, *Hydrobiologia,* Vol. 292-293, No. 1, pp. 1-7, ISSN: 0018-8158.

Koschan, A. & Abidi, M. (2008). *Digital Color Image Processing* (First Edition), John Wiley & Sons, Inc., ISBN 9780470147085, Hoboken, New Jersey.

Lerma-Aragón, J. & Álvarez-Borrego, J. (2009). Vectorial signatures for invariant recognition of position, rotation and scale pattern recognition, *Journal of Modern Optics*, Vol. 56, No 14, pp. 1598-1606, ISSN 0950-0340.

Obinata, G. & Morris, G. (2007). *Vision systems, segmentation and pattern recognition*, I-Tech Education and Publishing, ISBN 978-3-902613-01-1, Vienna, Austria.

Pech, J., Cristobal, G., Álvarez-Borrrego, J., and Cohen, L., (2001), Automatic system for phytoplanktonic algae identification, *Limnetica*. Vol. 20, No. 1, pp. 143-158, ISSN 0213-8409

Pratt, W. K. (2007). *Digital image processing*, (4th ed), John Wiley & Sons, Inc., ISBN: 978-0-471-76777-0, Hoboken, New Jersey.

Schwartz, E. (1994). Topographic mapping in primate visual cortex: anatomical and computation approaches, *Visual Science and Engineering. Models and Applications,* Ed. Marcel Dekker, New York. ISBN: 0824791851.

Soh, H. Y. & Suh, H.L. (2000). A new species of Acartia (Copepoda, Calanoida) from the Yellow Sea, *Journal of Plankton Research,* Vol. 22, (February 2000), pp. 321-337, ISSN: 0142-7873.

Solorza, S. & Álvarez-Borrego, J., (2010). Digital system of invariant correlation to position and rotation, *Optics Communications*, Vol. 283, No. 19 (October 2010), pp. 3613–3630, ISSN: 0030-4018

Solorza, S. & Álvarez-Borrego, J., (2011). Digital system of invariant correlation to position and scale using adaptive ring masks and unidimensional signatures, *Proceedings of SPIE*, 22nd Congress of the International Commission for Optics: Light for the

Development of the World, ISBN: 9780819485854, Vol. 8011, Puebla, México, August 2011.

Westland, S. & Ripamonti, C. (2004). *Computational Colour Science using MATLAB*, John Wiley & Sons Ltd, ISBN 0-470-84562-7, West Su.

15

Multidimensional Features Extraction Methods in Frequency Domain

Jesus Olivares-Mercado, Gualberto Aguilar-Torres, Karina Toscano-Medina, Gabriel Sanchez-Perez, Mariko Nakano-Miyatake and Hector Perez-Meana
National Polytechnic Institute
Mexico

1. Introduction

Pattern recognition have been a topic of active research during the 30 years, due to the high performance that these schemes presents, when they have been used in the solution of many practical problems in several fields of science, medicine and engineering. The efficiency of pattern recognition algorithms strongly depends in an accurate features extraction scheme that be able to represent the pattern under analysis using a number of parameters as small as possible, while keeping a large intra-pattern and very low inter-pattern similarities. These requirements have led to the development of several feature extraction methods, which can be divided in three groups. Feature extraction methods in time domain, spatial domain and frequency domain. In all cases the proposed feature extraction methods strongly depend of the specify applications. Thus the features extraction methods performing well in some applications, may do not perform well in others, for example the features extraction methods used for speech or speaker recognition are quite different to those used for fingerprints or face recognition. This chapter presents an analysis of some successful frequency domain feature extraction methods that have been proposed for applications involving audio, speech and images pattern recognition. Evaluation results are also provided to show the effectiveness of such feature extraction methods.

2. Feature extraction in speaker recognition

Speech is a widely used biometric feature for person recognition, where a person is recognized through his voice. To develop this kind of systems, several frequency domain methods have been proposed such as LPCepstral, described below, the LPCESPTRAL combined with dynamic features like the Cepstral Mean Normalization (CMN). In this section an analysis of this feature extraction methods is provided together with some evaluation results to show recognition performance.

2.1 Speaker recognition system

A general speaker recognition system, shown in Fig 1, consists mainly, of three stages: the feature extraction stage, where appropriate information is estimated in a suitable form and size, from the speech signal to obtain a good representation of the speaker features, the classifier stage, where the speaker models are adapted using the feature vectors, and the

decision stage, where the recognition decision is taken. The SRS system under analysis firstly extracts the features vector from the speaker voice. To this end, firstly estimates the LPC-Cepstral (Ganchev et al., 2002) coefficients using only the voiced parts of speech signals. Then using the estimated LPC-Cepstral coefficients, the dynamic features are estimated to enhance the speaker features vector. Next the estimated dynamical features vector is feed to a Gaussian Mixture Model, GMM, which is used to obtain a representative model for each speaker. Taking into account that a SRS can be improved taking the voiced part of the speech signal because this contains the main information relative to the speaker identity (El-Solh, 2007; Markov & Nakagawa, 1999; Plumper et al., 1999). For this reason in this book the features vector will be derived from the LPC-cepstral extracted only from the voiced part speech signal.

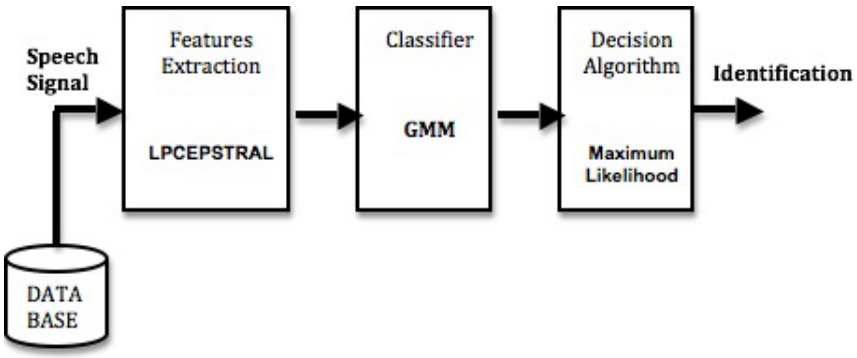


Fig. 1. General Speaker Recognition System

2.2 Feature vector extraction

A good performance of any pattern recognition system strongly depends on the extraction of a suitable feature vector that allow unambiguous representation of the pattern under analysis, with a number of parameters as small as possible. A simple way to estimate the speaker characteristics is the use of the linear prediction coefficients (LPC) of speech signal. The main reason about it is the fact that using these parameters can satisfactorily represent the structure of the vocal tract. However, it has been reported that better performance can be obtained if the LPC are combined with some frequency domain representation. One of these representations are the LPC features combined with the cepstral analysis, which allows to get a robust speaker characterization with low sensitivity to the distortion introduced in the signal transmitted through conventional communication channels (**?**).

The features vectors extracted from the whole speech signal provide a fairly good performance. However, when the LPCepstral coefficients are obtained from the LPC analysis, useful information of the speaker is still ignored or not taken in account, such as the pitch that is a specific feature of the individual speaker identity widely used to represent the glottal flow information.The performance of SRS can be seriously degraded when the SRS uses speech signal transmitted though some communication channel, such as a telephone one, due to the frequency response of the communication channel as well as the environment or the microphone characteristics. The LPCepstral coefficients have shown to be robust for

reducing the problem of low speech quality. In most SRS even if they have a good performance using the same data training (closed test), their performance considerably degrades when the systems are used with different data set (open test), because the data for closed and open test for each speaker may have different acoustic conditions. Thus, channel normalization techniques may have to be used to reduce the speaker features distortion, keeping in such way a good recognition performance. Among the channel normalization technique we have the Cepstral Mean Normalization (CMN), which can provide a considerable environmental robustness at a negligible computational cost (Liu et al., 1993). On the other hand, the use of more than one speaker feature is proposed as well as combination of them to get a more robust feature vector. To improve the SRS performance the LPCepstral coefficients obtained using the CMN can be combined with the pitch information because the pitch is a very important speaker feature.

2.2.1 Features vector derived form LPCepstral

To estimate the LPCepstral coefficients, firstly the speech signal is divided in segments of 20 ms length with 50% overlap using a Hamming window. Next, the LPC coefficients are estimated using the Levinson algorithm such that the mean square value of prediction error given by.

$$E[e(n)] = E\left[S(n) - \sum_{i=1}^{P} a_i S(n-i)\right] \tag{1}$$

becomes a minimum, where $E[.]$ is the expectation operator, P is the predictor order and a_i is the $i-th$ linear prediction coefficient (LPC). Next, once the LPC vector has been estimated, the LPCepstral coefficients can be obtained in a recursive way as follows (Simancas-Acevedo et al., 2001).

$$c_n = -a_n + \frac{1}{n}\sum_{i=1}^{n}(n-i)a_i c_{n-i}, \qquad n > 0 \tag{2}$$

where c_n is the $n-th$ LPCepstral coefficient. Thus the SRS feature vector becomes

$$X_t = \left[c_{1,t}, c_{2,t}, c_{3,t}, \cdots, c_{d,t}\right] \tag{3}$$

where t denotes the frame number.

2.2.2 Features vector derived from LPCepstral of voiced segments

The pitch and voiced part detection plays a very important roll in the speaker recognition systems because the pitch value and the voiced segments of speech signals contain the most important information about the speaker identity. Then the feature vector could be extracted only from the voiced segments of speech signal (Rabiner & Gold, 1975). To this end, firstly the pitch period is detected using the autocorrelation method (Campbell, 1997) as follows: Initially, the speech signal is segmented in frames of 20 ms with 10 ms overlap using a Hamming window; next the center clipper method (Ganchev et al., 2002) is applied to the windowed frame to reduce the effect of the additive noise intrinsic within of the speech signal. Subsequently, the autocorrelation of the center clipped segment is obtained. Finally the pitch value is estimate as the distance between two consecutive positions in which the normalized autocorrelation sequence is larger than a given threshold, as proposed in

(Seung-Jin et al., 2007). Using the pitch information, the speech segment is then classified as voiced or unvoiced, because the pitch only appears in the voiced segments. Thus, if the pitch does not exist the speech segment is considered as a unvoiced segment (Campbell, 1997; **?**); and the pitch exists then speech segment is classified as a voiced speech segment. The Fig. 2 shows clearly the result of this procedure that proves that the detection of the voiced part is correctly done.

If only the voiced segments of speech signal are taken in account for features extraction, the original speech signal is transformed into a new speech signal containing only voiced parts, neglecting in such way the unvoiced and noisy silence parts, as shown in Fig. 3. This may improve the feature extraction because the unvoiced and silence parts provide non-useful information (Pool & Preez, 1999).

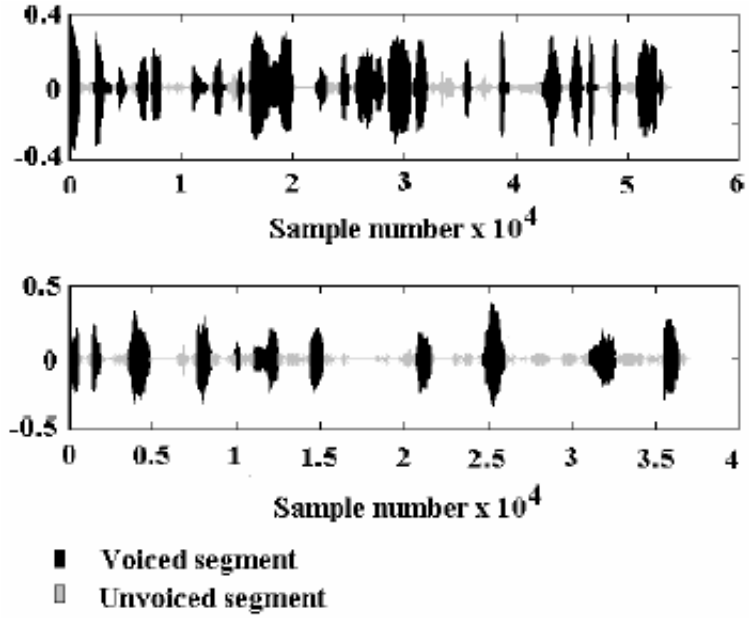


Fig. 2. Pitch period detection in two speech signals

The new signal, with only the voiced parts, has less samples number than the original one, but contains the essential parts required to estimate the principal features that identify the speaker and, as shown in the Fig. 3, the number of data is reduced, in many cases, as far as 50%. Once the new signal is constructed only with the voiced parts, it is divided in segments of 20 ms length with 50% overlap using a Hamming window. Next, the LPC coefficients are estimated using the Levinson algorithm as mentioned above, and then the LPCepstral coefficients are estimated using the eq. (2). Here two different features vectors can be estimated. The first one consists only of the LPCepstral of voiced segment.

$$X_t^v = \left[c_{1,t}^v, c_{2,t}^v, c_{3,t}^v, \cdots, c_{d,t}^v\right] \tag{4}$$

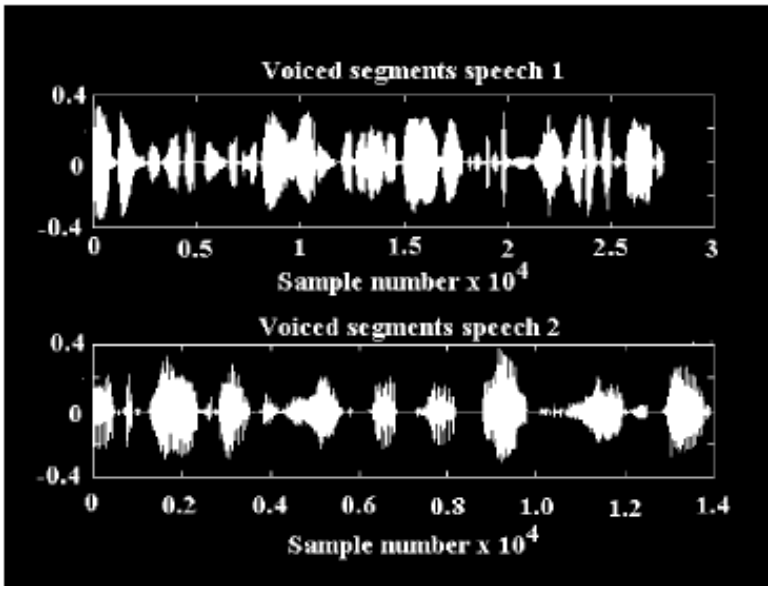


Fig. 3. Speech signal reduction taking only voiced part

and the second one consists of LPCepstral coefficients and pitch information

$$X_t^p = \left[c_{1,t}^v, c_{2,t}^v, c_{3,t}^v, \cdots, c_{d,t}^v, log_{10}F_{0,t}\right] \tag{5}$$

where $c_{n,t}^v$ is the $n-th$ LPCepstral coefficient and $F_{0.t}$ is the inverse of the pitch period at the block t. Here $log_{10}F_{0,t}$ is used instead of the pitch period, because the probability distribution of the $log_{10}F_{0,t}$ is close to the normal distribution.

2.2.3 Reinforcing and enhancing feature vectors

In long distance speaker recognition, the speech signal is transmitted through a channel communication and then is processed by the SRS. However, the speech signal suffers some distortion or variation due to the communication channel effects, noise environment, etc. Because these distortions are added to the principal components of the speech signal, it is necessary to remove the undesirable information before to proceed with the recognition process. To this end it would be convenient to enhance the estimated feature vector. Thus we can subtract the global average vector from all feature vector components. In this process, known as Cepstral Mean Normalization (CMN) (Murthy et al., 1999; Reynolds, 1995; Hardt & Fellbaum, 1997), it is assumed that the mean values of LPCepstral coefficients of clean speech is zero, so that if the mean value is subtracted from the feature vector components its mean value becomes zero. This avoids the distortion introduced by the additive noise when the signal passes through the communication channel. This technique is equivalent to a high-pass filtering of LPCepstral coefficients, because the CMN estimates the mean value of the LPCepstral vector coefficients and subtracts it from each component, as shown in eq. (6):

$$CMN_{n,t} = c_{n,t} - \frac{1}{T}\sum_{t=1}^{T} c_{n,t}, \qquad 1 \leq n \leq d \tag{6}$$

where $c_{n,t}$ is $n-th$ LPCepstral coefficient at block t and T is the total number of frames in which the speech signal was divided to extract the feature vectors. Fig. 4a and Fig. 4b show the effect produced in the feature vector when the Cepstral Mean Normalization (CMM) technique is applied to one of the LPCepstral coefficients extracted from the only the voiced part of the speech signal. In this situation the features vector becomes

$$CMN_t = \left[CMN_{1,t}, CMN_{2,t}, CMN_{2,t}, \cdots, CMN_{d,t}\right] \tag{7}$$

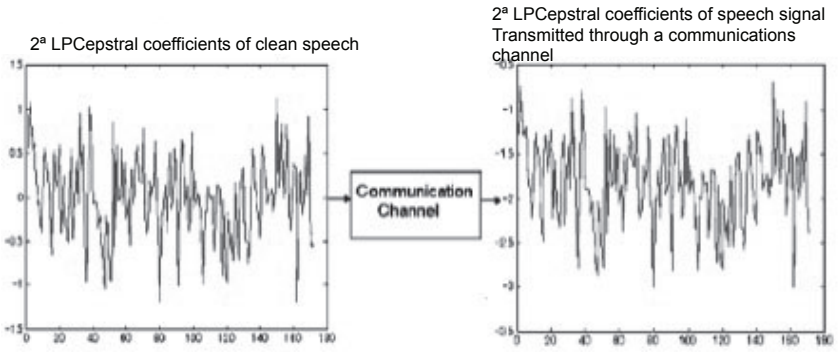


Fig. 4. The second LPC-Cepstral coefficient extracted from the speech signal

2.3 Results

Here present some results of a SRS that were evaluated using a feature vector with 16 LPCepstral coefficients extracted from only voiced part of the speech signals. The first evaluation of the baseline system is using these 16 LPCepstral coefficients extracted from only voiced part. The second evaluation is using the CMN technique to enhance the feature vector quality which has been affected by the channel communication and the environment noise. The third evaluation is using a combination of LPCepstral and pitch information and the forth evaluation is using the combination of LPCepstral coefficients and pitch information applying the CMN technique. All system evaluations, which are discussed in each respective section, are presented in the Table 1 and Table 2 where for system evaluation in close test, the same data for training was used and for system evaluation in open test, 2 different repetitions were used which were stored in different times, giving a total of 3658 phrases..

3. Feature extraction in face recognition

Face recognition has a large amount of biometric applications. The intra-person variations of the face image derive mainly from changes in facial expressions, illumination conditions as well as because the use of some accessories such as eyeglasses and muffler, etc. These

Feature Vector	LPCepstral From whole speech signal	LPCepstral From voiced part
Evaluation(Close test) 7147 phrases	96.61%	97.13%
Evaluation(Open test) 3658 phrases	82.34%	83.57%

Table 1. Results using whole and voiced part speech signal

Feature Vector	LPCepstral From voiced part	LPCepstral From voiced parts using CMN	LPCepstral From voiced and pitch information	LPCepstral From voiced parts using pitch and CMN
Closed test 6581 phrases	93.31%	80.72%	99.18%	97.29%
Open test 3282 phrases	76.97%	70.88%	80.29%	77.57%

Table 2. Results with different features vectors

variations make the face recognition a very difficult task. Most approaches to face recognition are in the image domain whereas we believe that there are more advantages to work directly in the spatial frequency domain. By going to the spatial frequency domain, image information gets distributed across frequencies providing tolerance to reasonable deviations and also providing graceful degradation against distortions to images (e.g., occlusions) in the spatial domain.

3.1 Face recognition system

This section provides a detailed description of a general face recognition system which consists in three stages each one. Figure 5 shows the block diagram of this system. Firstly in the pre-processing the input image is normalized, equalized or some other method for enhance a image. In the feature extraction the phase spectrum is extracted, after that, the Principal Components Analysis (PCA) (Kriegman et al., 1997; Hager et al., 1999) is applied to the phase spectrum to obtain a dominant feature of the faces. Next, the features in principal components space are fed into classifier and the image will be classified to the class given maximum likelihood.

3.2 Feature vector extraction

In the method proposed by Savvides (Savvides et al., 2004) were considering the combination of principal component analysis and phase spectrum of an image. Oppenheim (Oppenheim et al., 1980; Lim & Oppenheim, 1981) show that the image phase spectrum contains the most important information required for face image recognition, doing less relevant the use of the magnitude spectrum. His research also shows that getting only the phase spectrum of an image, can reconstruct the original image to a scale factor, therefore, information phase is the most important in the representation of a 2D signal in the Fourier domain. This is also demonstrated by a simple experiment, this is shown in Figure 6, in which the face of A (a) is reconstructed using the phase spectrum of A (c) and the magnitude spectrum of face B (f);

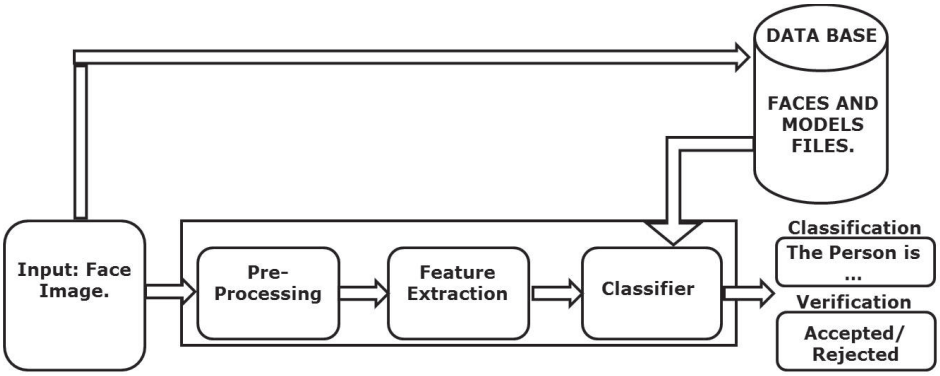


Fig. 5. General Face Recognition System

and the face of B (e) is reconstructed using the phase spectrum of B (g) and the magnitude spectrum of A (b). The reconstructed images, figures 2(d) and 2(h), show that the synthesized face image clearly resemble A (a) and B (e), respectively.

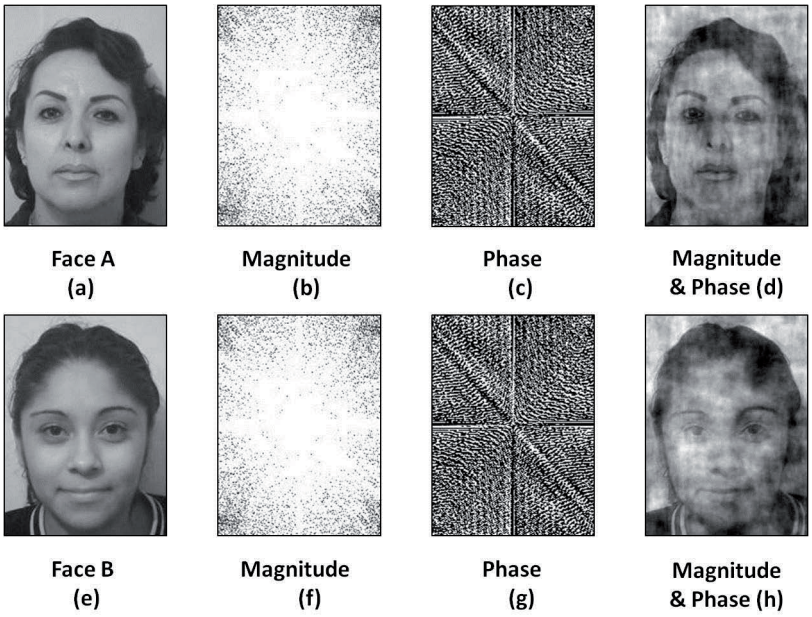


Fig. 6. Oppenheim Experiment

However, the performance of PCA in the frequency domain alone does not constitute any progress, this is because the eigenvectors obtained in the frequency domain are simply the Fourier transform of spatial domain. We begin this derivation by defining the standard 2-D discrete Fourier transform (DFT). Given an input 2-D discrete signal $x[m,n]$ of size $M \times N$

denote its Fourier transform as $X[k,l]$ whose Fourier transform pair is defined as follows:

$$x[m,n] \rightleftarrows X[k,l] \tag{8}$$

$$X[k,l] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n] exp\frac{-i2\pi km}{M} exp\frac{-i2\pi ln}{N} \tag{9}$$

$$X[m,n] = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} X[k,l] exp\frac{-i2\pi km}{M} exp\frac{-i2\pi ln}{N} \tag{10}$$

where $i = \sqrt{-1}$

If we have N training images then the covariance matrix of the Fourier transform is obtained by:

$$\widehat{\Sigma_f} = \frac{1}{N}\sum_{i=1}^{N}\{F(x_i - \widehat{\mu}\}\{F(x_i - \widehat{\mu}\}^T = F\widehat{\Sigma_s}F^{-1} \tag{11}$$

once the PCA was normalized, the eigenvectors w_f de $\widehat{\Sigma_f}$ are obtained by:

$$F\widehat{\Sigma_s}F^{-1}w_f = \lambda w_f \tag{12}$$

multiplying both sides by F^{-1} have:

$$\widehat{\Sigma_s}F^{-1}w_f = \lambda F_{-1}w_f \tag{13}$$

now formulate the space domain PCA and it shows that the space in the domain of the covariance matrix is:

$$C_s = \widehat{\Sigma_s} \tag{14}$$

$$C_s v_s = \lambda v_s \tag{15}$$

$$\widehat{\Sigma_s} v_s = \lambda v_s \tag{16}$$

where v_s is the space in the domain of the eigenvectors.

Comparing equation 13 with equation 16 we can see that is a relationship between space and frequency domain of the eigenvectors $(v_s and v_f)$ related by an inverse Fourier transform as follows:

$$v_s = F^{-1}v_f \tag{17}$$

3.3 Results

Here present some results, where the Face Recognition System was evaluated using the "AR Face Database", which has a total of 9,360 face images of 120 people (65 men and 55 women). This database includes 78 face images of each people with different illuminations, facial expression and partial occluded face images with sunglasses and scarf. To evaluate the performance of the proposed methods under several illumination and occlusion conditions, two different training sets are used. The first one consists of 1200 images, 10 images of each person, with different illumination and expression conditions; while the second set consists of

1200 face images, 10 per each person, with different illumination conditions, expressions and occlusions, which are the result of using sunglasses and scarf. The remaining images of the AR face database are used for testing. The table 3 show some results of face identification and table 4 some results of face verification.

	Training set 1	Training set 2
Eigenphases	80.63	96.28

Table 3. Results using Eigenphases for identification.

	False acceptance	False reject
Training set 1	0.5	19.07
Training set 2	0.5	3.80

Table 4. Results using Eigenphases for verification

4. Feature extraction in sound recognition

Sound recognition (Goldhor, 1993) has a large amount of application civil as well as military applications such as engine diagnostic, airplane or ship recognition, etc. This applications also depends on the above mentioned feature extraction methods. This section describes a sound recognition system using a frequency domain feature extraction method like LPCepstral.

4.1 Sound recognition system

Figure 7 shows the proposed system. This system consist of four sequential processes: first a common database of environmental sounds is obtained, after this a segmentation algorithm is applied to each token (file) of this database; third LPC-Cepstral features are extracted from each segmented file and the DFT is computed from these coefficients, finally the DFT magnitude is computed and a training strategy is adopted. The decision is taken at the final process and a recognition percentage is computed.

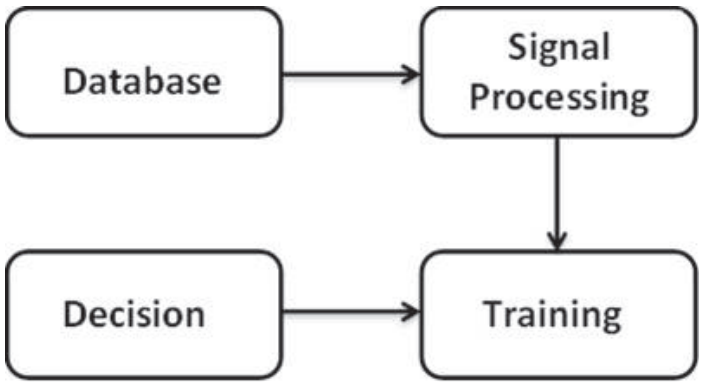


Fig. 7. General Sound Recognition System

4.2 Feature vector extraction

Figure 8 shows the applied processes in the signal analysis. With this signal analysis a high efficiency of feature extraction is obtained, this facilitates to the neural network the recognition process, this means that higher percentages of verification and identification can be obtained.

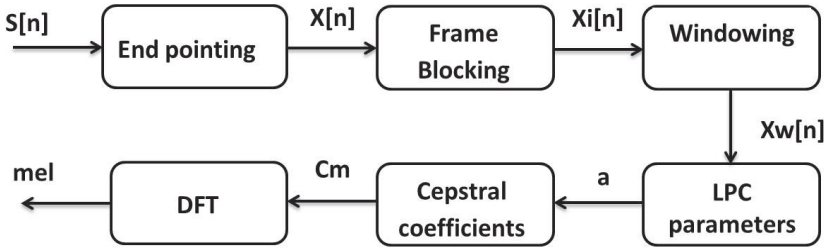


Fig. 8. Methodology used in the signal processing

4.2.1 End-pointing algorithm

In time domain, magnitude, energy, power, maximums and minimums can be computed from which, the energy is used. Once the energy was calculated, a reference is obtained, with this reference the signal can be limited. In the discrete case the energy is defined as:

$$E[n] = \sum_{n=-\infty}^{\infty} s^2[n] \tag{18}$$

Now, a gamma constant is defined, this constant indicates the number of samples taken from the signal. The following step is to make a relationship between the sound signal and the gamma constant:

$$E[n] = [(1-\gamma) * E_{n-1}] * [\gamma * y_n^2] \tag{19}$$

To each file stored in the database an end-pointing algorithm was applied. In order to limit the signal two thresholds of 20 and 10 the maximum energy must be defined, this corresponds to the percentage taken form the signal. This algorithm compares the thresholds with each sample of the energy until a sample is greater or equal to these thresholds, indicating the signal is beginning.

4.2.2 Frame blocking and windowing

The sound signal, $\widehat{S}[n]$, is blocked into frames of 240 samples that corresponds to 30 msec, in which voice is considered stationary (Kitamura & Hayahara, 1988), with adjacent frames being separated by 120 samples. The use of frames implies three parameters: frame size, frame increment and frame overlapping:

$$S_f = I_f + O_f \tag{20}$$

where I_f represents the frame beginning and O_f the frame overlapping.

To the sequence of analysis frames generated from each end-pointed file was applied a windowing algorithm, Hamming window was used:

$$\widehat{S}_w[n] = \widehat{S}[n]W[n] \tag{21}$$

where $0 < n < N-1$, N is the number of samples in the analysis frame and $W[n]$ is a Hamming window.

4.2.3 LPC and LPC-Cepstral parameters

In each window 17 LPC coefficients were calculated with Levinson-Durbin recursion. LPC-Cepstral coefficients can be derived directly from the set of LPC coefficients using the algorithm:

$$C[n] = -a[n] - \frac{1}{n}\sum_{k=1} n = 1kC[k]a[n-k] \tag{22}$$

where $n > 0$, $C_0 = a_0 = 1$, $k > p$ and $a[n]$ represents the linear prediction coefficients, estimated, using a linear filter, as it is shown in Figure 9 , where a given sound sample cam be approximated or predicted as a linear combination of its past p samples, as shown in eq. 22. The number of frames generated for each signal was of 64. The result in effect was that each signal was represented by a 17 by 64 array of the Cepstral coefficients, with the 64 rows representing time and the 17 columns representing frequency.

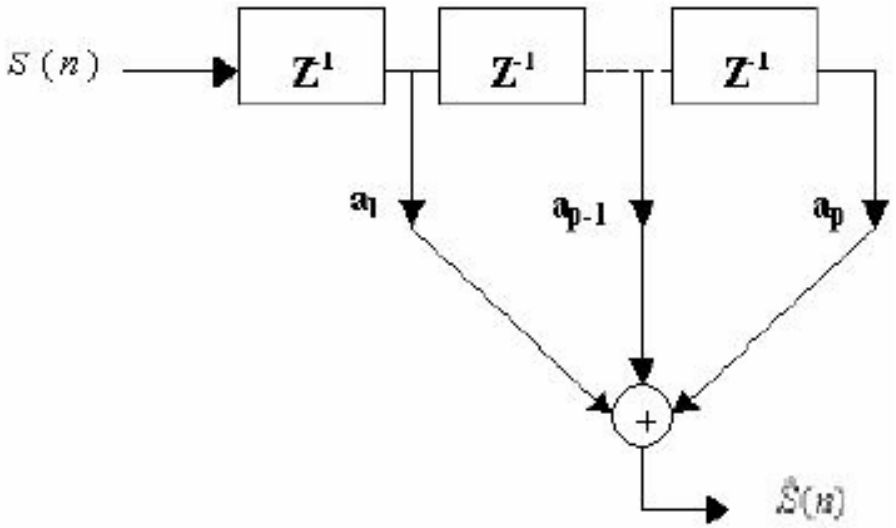


Fig. 9. Linear Prediction Filter used for the LPC-Cepstral Estimation

$$\widehat{S}(n) = \sum_{i=1}^{P} a(i)S(n-i) \tag{23}$$

where $S(n)$ is the estimated signal in time n, P is the filter order and a is the filter coefficients vector.

A 64-point DFT was then calculated for each column in the matrix and the first 32 points of this symmetrical transform retained. The resulting square matrix is a two dimensional Cepstral representation of the input signal. Each column corresponds to a particular spectral frequency, and ach row corresponds to a temporal frequency. The first column contains the DFT of the power envelope of the signal. The first row contains the DFT of the average signal spectrum. The first element of the first column contains the average signal power level. It Šs typical of a two-dimensional Cepstral representations of acoustic signals, and certainly for our signals, that this corner element is the largest component and the size of the components in the first row and first column are larger than the size of interior matrix components. After this the DFT magnitude for each column in the matrix is computed. The LPC-Cepstral coefficients, which are the Fourier transform representation of the spectrum, have been shown to be more robust for speech recognition than the LPC coefficients, in this case we applied this method. The Cepstral transforms were calculated for two variants of the spectrum within each frame: a linear variant and one in which the frequency scale was warped using mel frequency transformation.

4.3 Results

The used model is an artificial neural network backpropagation. The traditional error backpropagation algorithm is used. For each sound pattern, 50 sound files were used in the network training process. The sound samples are first normalized so that the average magnitude becomes zero and the standard deviation is one. Clusters, or classes, were formed by grouping the feature vectors for each type of sound. For the network training, the ideal number of hidden-layer neurons was chosen from the experimental work. The hope, of course, is that the samples of each sound will cluster together in that space and that cluster for different sounds will be rejected.

Two neural networks, per sound-source, were trained. Four stages (one per neural network) were necessary for the network training and each stage corresponds to each file stored in the database. 32 input-layer neurons were necessary for the neural network training, 10, 15 and 20 hidden-layer neurons were used in this neural network and the best results were obtained with 20 neurons; 1 output-layer neuron, per network, was necessary for verify the source-sounds and 4 output-layer neurons in the identification process.

5. Feature extraction in fingerprint recognition

Fingerprint Identification (Aguilar et al., 1994) is one of the most reliable and popular personal identification methods. The performance of minutiae extraction algorithms and other fingerprint recognition techniques relies heavily on the input fingerprint images quality. In an ideal fingerprint image, ridges and valleys alternate and flow in a locally constant direction. In such situations, the ridges can be easily detected and the minutiae can be accurately located in the image. However, in practice, because of skin conditions (e.g., wet or dry, cuts, and bruises), sensor noise, incorrect finger pressure; and inherently low-quality fingers (e.g., elderly people, manual workers), etc. a significant amount of fingerprints are poor quality images.

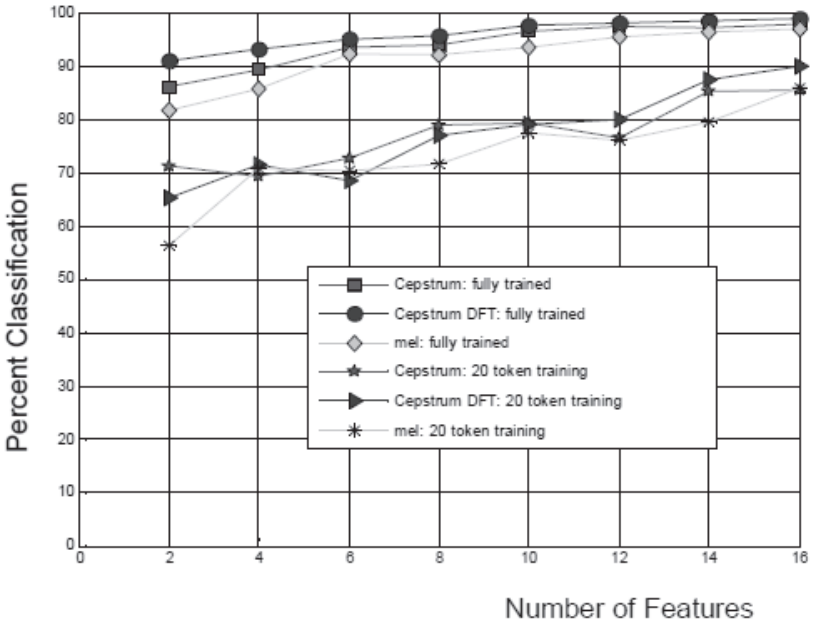


Fig. 10. Percent of correct classification of test.

5.1 Fingerprint recognition system

Figure 10 shows the main stages of a biometric system. As mentioned above, a good quality of the input image ensures that the recognition will be higher. Therefore, we can use Fourier to enhance the quality of input images of a biometric system.

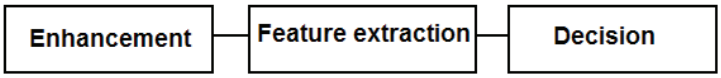


Fig. 11. General Fingerprint Recognition System

5.2 Feature vector extraction

A good fingerprint image will have high contrast and well-defined ridges and valleys, and a poor quality fingerprint will have low contrast and ridges and valleys poorly defined. In a biometric system we can have images with different qualities, as shown in Figure 11. Thus the goal of an enhancement algorithm is to improve the clarity of the ridge structures in the recoverable regions and classify the unrecoverable regions as too hazy for further processing. Because of the regularity and continuity properties of the fingerprint images, the occluded and corrupted regions can be recovered using the contextual information from the surrounding neighborhood. This section describes a contextual filtering completely in the frequency domain, where each image is convolved with a precomputed filter whose size is

equal to the image size. Therefore the algorithm does not use the full contextual information provided by the fingerprint image.

Fig. 12. Set of Fingerprints

An advantage of this approach is that, for its operation, it does not require the computation of intrinsic images; which has the effect of increasing the dominant spectral components while attenuating the weak ones. However, in order to preserve the phase, the enhancement also retains the original spectrum F(u,v). In this algorithm the fingerprint enhancement are transformed from spatial domain to frequency domain by Fourier transforming.

5.2.1 Fourier domain filtering

Sherlock (Sherlock et al., 1994) perform contextual filtering completely in the frequency domain, where each image is convolved with a precomputed filter whose size is equal to the image size. However, the algorithm assumes that the ridge frequency is constant through out the image in order to prevent having a large number of precomputed filters. Therefore the algorithm does not use the full contextual information provided by the fingerprint image. Watson (Watson, Candela & Grother) proposed another approach for performing image enhancement completely in the frequency domain, which is based on the root filtering technique. In this approach the image is divided into overlapping blocks, where in each block the enhanced image is obtained as:

$$I_{enh}(x,y) = FFT^{-1}\left[F(u,v)|F(u,v)|^{k}\right] \tag{24}$$

$$F(u,v) = FFT(I(x,y)) \tag{25}$$

where the k in formula (24) is an experimentally determined constant, which we choose equal to 0.45. Here, while a higher "k" improves the appearance of the ridges, filling up small holes in ridges, a too large value of "k" may result in false joining of ridges, such that a termination might become a bifurcation. Another advantage of this approach is that, for its operation, it does not require the computation of intrinsic images; which has the effect of increasing the dominant spectral components while attenuating the weak ones. However, in order to preserve the phase, the enhancement also retains the original spectrum $F(u,v)$. Figure 12 shows fingerprint images after of the FFT.

5.3 Results

In this section a fingerprint recognition algorithm using FFT was evaluated. The tests consisted of the recognition of 125 people. Table 5 shows the recognition results obtained using FFT. We show that it results in 11.3% improvement in recognition rate over a set of 1000

Fig. 13. Set of Fingerprints

images. True recognition means that the person was effectively recognized, the rest is divided into two: false recognition and without recognition. False recognition occurs when a person is confused and without recognition when the system does not deliver any possible identified person.

Total percentage	True recognition	False recognition	without recognition
Without FFT	81.8%	9.4%	8.8%
Using FFT	93.1%	3.8%	3.1%

Table 5. Test results made to 1000 images

The ROC curves before and after enhancement are as shown in the Figure 13.

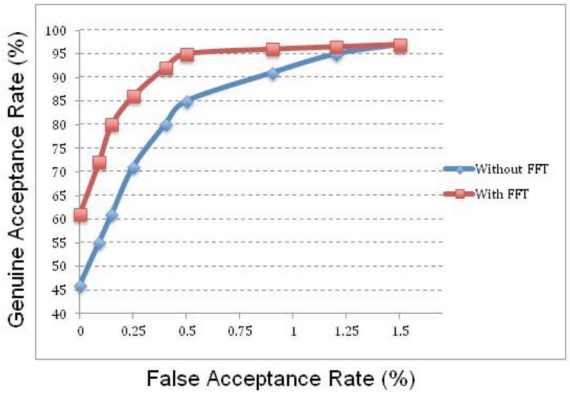


Fig. 14. Set of Fingerprints

6. References

Ganchev T., Tsopanoglou A., Fakotakis N., Kokkinakis G. (2002). Probabilistic Neural Networks Combined with GMMs For Speaker Recognition over Telephone Channels. 14-th International Conference On Digital Signal Processing (DSP 2002), Santorini, Greece, Volume II, 1081Ű1084.

El-Solh A., Cuhadar A., Goubran R. A., (2007). Evaluation of Speech Enhancement Techniques for Speaker Identification in Noisy Environments. Multimedia Workshops, 2007. ISMW '07. Ninth IEEE International Symposium on, page(s): 235 - 239 Print ISBN: 9780-7695-3084-0.

Markov K. P., Nakagawa S. (1999). Integrating Pitch and LPC-Residual Information with LPC-Cepstral for Text-independent Speaker Recognition. J. Acoustic Society of Japan (E), 20, 4, 281-291.

Plumper M. D., Quatieri T. F., Reynolds D. A. (1999). Modeling of the Glottal Flow Derivative Waveform with Application to Speaker Identification. IEEE Transactions on Speech and Audio Processing, vol. 7, Nř5.

Liu F., Stern R. M., Huang X., Acero A. (1993). Efficient Cepstral Normalization For Robust Speech recognition. Proceeding HLT '93 Proceedings of the workshop on Human Language Technology. ISBN:1-55860-324-7, Stroudsburg, PA, USA.

Simancas-Acevedo E., Kurematsu A., Nakano-Miyatake M., Perez-Meana H. (2001). Speaker Recognition Using Gaussian Mixtures Model. Lecture Notes in Computer Science, Bio- Inspired Applications of Connectionism, Springer Verlag, Berlin, 287Ű294.

Rabiner B., Gold B. (1975). Digital Processing of Speech Signals. Prentice Hall, Englewood Cliffd, NJ.

Campbell J. P. (1997). Speaker Recognition: A Tutorial. Proceedings of the IEEE, vol. 85, Nř9, 1437Ű1462.

Seung-Jin Jang, Seong-Hee Choi, Hyo-Min Kim, Hong-Shik Choi, Young-Ro Yoon, (2007). Evaluation of Performance of Several Established Pitch Detection Algorithms in Pathological Voices. Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE , page(s): 620 - 623, ISSN : 1557-170X, Print ISBN: 978-1-4244-0787-3 .

Pool J., du Preez J. A. (1999). HF Speaker Recognition. Thesis notes, Digital Signal Processing Group, Department of Electrical and Electronic Engineering, University of Stellenbosch, March 1999.

Murthy H. A., Beaufays F., Heck L. P., Weintraub M. (1999). Robust Text-Independent Speaker Identification over Telephone Channels. IEEE Transactions on Speech and Audio Processing, vol. 7, Nř5.

Reynolds D. A. (1995). Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. IEEE Transactions on Speech and Audio Processing, vol. 3, Nř1, 72Ű83.

Hardt D., Fellbaum K. (1997). Spectral Subtraction and Rasta Filtering in Text Dependent HMM-based Speaker Verification. Proc. of ICASSP, vol. 2, 867-870.

Kriegman D., Belhumeur P., Hespanha J. (1997). Eigenfaces vs fisherfaces: recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7), 711-720.

Hager J., Ekman P., Sejnowski T., Donato G., Bartlett M. (1999). Classifying facial actions. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, 974-989.

Savvides M., Kumar B.V.K.V., Khosla P.K. (2004). Eigenphases vs. eigen-faces. Pro- ceedings of the 17th International Conference on Pattern Recognition, Vol. 3, 810-813.

Oppenheim A. V., Hayes M. H., Lim J. S. (1980). Signal reconstruction from phase or magnitude. IEEE Trans. Acoust., Signal Processing, Vol. 28, pp. 672-680.

Lim J. S., Oppenheim A. V. (1981). The importance of phase in signals. Proc. IEEE, 69(5), 529-541.

Goldhor, R.S., (1993), Recognition of Environmental Sounds, Proceedings of ICASSP, Vol. 1, 149-152.

Kitamura T., Hayahara, E. (1988), Word Recognition Using a Two-Dimensional Mel-Cepstrum in Noisy Enviroments, paper PPP6 presented at the 2nd Joint Meeting of the ASA and ASJ, Hawwaii.

Aguilar, G., Sanchez, G., Toscano, K., Salinas, M., Nakano, M., Perez, H., (2007), Fingerprint Recognition, Internet Monitoring and Protection, 2007. ICIMP 2007. Second International Conference on, page(s): 32, Print ISBN: 0-7695-2911-9 .

Sherlock B. G., Monro D. M., Millard K., (1994), Fingerprint enhancement by directional Fourier filtering, in: Visual Image Signal Processing, Vol. 141, pp. 87Ð94, 1994.

Watson C.I., Candela G.I., and Grother P.J., (1994), ÒComparison of FFT Fingerprint Filtering Methods for Neural Network Classification,Ó Tech. Report: NIST TR 5493, Sept. 1994.