

Design of Non-integer Order Digital Differentiator

*A dissertation submitted in partial fulfillment of the
requirement for the award of degree of*

Master of Technology

in

VLSI Design



Submitted by

Md. Tarique Siddique

Roll No. 601161032

Under the supervision of

Mr. Sanjay Kumar

Assistant Professor, ECED

Thapar University, Patiala

Department of Electronics & Communication Engineering


Thapar University, Patiala – 147004

DECLARATION


I hereby declare that the work which is being presented in the dissertation entitled, "Design of Non-integer Order Digital Differentiator" in partial fulfilment of the requirement for the award of degree of Master of Engineering in VLSI Design submitted in Electronics and Communication Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Mr. Sanjay Kumar, Assistant Professor, ECED and refers other researcher's work which are duly listed in the reference section.

The matter presented in this dissertation has not been submitted in any other University/Institute for the award of degree.

Date: 27/06/13


Md. Tarique Siddique
Roll No: 601161032


It is certified that the above statement made by the student is correct to the best of my knowledge and belief.


28/06/13
Mr. Sanjay Kumar
Assistant Professor
ECED, Thapar University

Countersigned by:



Head
ECED, Thapar University
Patiala-147004


Dean of Academic Affairs
Thapar University
Patiala- 147004

ACKNOWLEDGEMENT

First of all, I would like to express my gratitude to **Mr. Sanjay Kumar, Assistant Professor**, Electronics and Communication Engineering Department, Thapar University, Patiala for his patient guidance and support throughout my work. I am truly very fortunate to have the opportunity to work with him. I found his guidance to be extremely valuable.

I am also thankful to **Professor Dr. Rajesh Khanna, Head of the Department**, Electronics and Communication Engineering Department, entire faculty and staff of Electronics and Communication Engineering Department. I would also like to thank my friends who devoted their valuable time and helped me in all possible ways towards successful completion of this work. I thank all those who have contributed directly or indirectly to this work.

Lastly, I would like to thank my parents for their unconditional support and encouragement.

Md. Tarique Siddique

Roll no. 601161032

ABSTRACT

The design of digital filters at low frequencies range has become increasingly important as it can be used to design all types of filters. Use of Integer order calculus for the purpose of design often results in narrow bandwidth for the low pass. With the development of fractional order calculus in recent years the response becomes more ideal. But the drawback of design is that due to multiple variables, the response is bit slow.

The major objective of this work is to understand the different design strategy for digital differentiators and compare their response for various orders and to explore new design techniques for designing fractional order IIR differentiator. A stable minimum phase, second-order, low-pass IIR digital differentiators is developed by inverting the transfer functions of a class of second-order integrators, stabilizing the resulting transfer functions, and compensating their magnitudes.

The class of second-order integrators are obtained by interpolating the traditional Simpson and trapezoidal integrators. The designed digital differentiator is modelled to find the correct response by passing some test signal. The designed differentiators extend the frequency range of operation beyond that possible by using either of the two traditional integrators. The low order and high accuracy of the filters make them attractive for real time applications.

Al-Alaoui operator and new mapping function (NMF)) models are used for the design of fractional order digital differentiator. The proposed low-pass differentiators are shown to have shorter transition regions, and thus better ability to suppress high frequency noise, for much lower order filters.

TABLE OF CONTENTS

	Page No.
DECLARATION	i
ACKNOWLEDGEMENT	ii
ABSTRACT	iii
TABLE OF CONTENTS	iv
LIST OF FIGURES	vi
LIST OF TABLES	viii
ABBREVIATIONS	ix
Chapter 1 INTRODUCTION	1-5
1.1 FILTER	1
1.2 TYPES OF FILTERS	1
1.2.1 Analog Filters	1
1.2.2 Digital Filters	2
1.3 ADVANTAGES OF USING DIGITAL FILTERS	3
1.4 APPLICATIONS OF DIGITAL FILTERS	3
1.5 ORGANIZATION OF THE DISSERTATION	5
Chapter 2 LITERATURE REVIEW	6-15
Chapter 3 DIGITAL FILTERS	16-30
3.1 OVERVIEW	16
3.2 IMPULSE RESPONSE	18
3.3 FILTER CHARACTERISTICS	19
3.3.1 HOW INFORMATION IS REPRESENTED IN SIGNALS	19
3.3.2 TIME DOMAIN PARAMETERS	20
3.3.3 FREQUENCY DOMAIN PARAMETERS	22
3.4 TYPES OF DIGITAL FILTERS	26
3.4.1 IIR FILTERS	26
3.4.2 FIR FILTERS	27
3.5 UNDERSTANDING POLE ZERO PLOT AND STABILITY ANALYSIS	29
Chapter 4 IIR DIGITAL FILTERS	31-41
4.1 IMPULSE INVARIANCE	31
4.2 BILINEAR TRANSFORMATION	32
4.3 TRANSFER FUNCTION OF AN IIR FILTER	34
4.4 IIR FILTER TYPES	35

	4.4.1	Butterworth Filters	35
	4.4.2	Chebyshev Filters	37
	4.4.3	Chebyshev II filters or Inverse chebyshev filters	38
	4.4.4	Elliptic Filters	39
4.5		SELECTING A DIGITAL FILTER DESIGN	41
Chapter 5		MATHEMATICAL MODELLING OF IIR DIGITAL FILTERS	42-48
5.1		INTEGER ORDER IIR DIGITAL DIFFERENTIATOR	43
	5.1.1	BASIC INTEGRATION RULES	43
		5.1.1.1 Simpson Integrator	43
		5.1.1.2 Trapezoidal Integrator	44
	5.1.2	INTEGRATOR DESIGN	45
	5.1.3	IIR-TYPE INTEGER ORDER DIGITAL DIFFERENTIATOR DESIGN	46
5.2		IIR-TYPE FRACTIONAL ORDER DIGITAL DIFFERENTIATOR	46
Chapter 6		SIMULATION RESULTS	49-57
6.1		INTEGER ORDER IIR DIGITAL DIFFERENTIATOR DESIGN	49
6.2		FRACTIONAL ORDER IIR DIGITAL DIFFERENTIATOR DESIGN	54
Chapter 7		CONCLUSION & FUTURE SCOPE	58-59
7.1		CONCLUSION	58
7.2		FUTURE SCOPE	59
		PUBLICATIONS	60
		REFERENCES	61-63

LIST OF FIGURES

Figure 1.1	General filtering method	1
Figure 1.2	Analog implementation of Filter	2
Figure 1.3	Block diagram of Digital Filter	2
Figure 3.1	(a) Impulse response	17
	(b) Step response	17
	(C) Frequency response on linear scale	17
	(d) Frequency response on decibel scale	17
Figure 3.2	Impulse Response	18
Figure 3.3	(a) Transition time of poor filter	21
	(b) Transition time of good filter	21
	(c) Overshoot of poor filter	21
	(d) Overshoot of good filter	21
	(e) Phase linearity of poor filter	21
	(f) Phase linearity of good filter	21
Figure 3.4	Various types of frequency responses of filters	22
Figure 3.5	Parameters for evaluating frequency domain parameters	23
Figure 3.6	(a) Impulse response of low pass filter	25
	(b) Frequency response of low pass filter	25
	(c) Impulse response of high pass filter	25
	(d) Frequency response of high pass filter	25
Figure 3.7	General IIR filter structure	26
Figure 3.8	General FIR filter	28
Figure 4.1	Comparison of S and Z plane using impulse invariance method	32
Figure 4.2	Comparison of S and Z plane using bilinear transformation technique	34
Figure 4.3	Response of Butterworth Filter	36
Figure 4.4	Response of Chebyshev filter	37
Figure 4.5	Response of Inverse Chebyshev filter	39
Figure 4.6	Response of Elliptic Filter	40
Figure 4.7	Flowchart for selecting filter	41
Figure 5.1	Deriving Simpson's rule	44
Figure 5.2	Deriving Trapezoidal rule	45
Figure 6.1	Magnitude response of obtained differentiator lying between simpson	49

	and trapezoidal response	
Figure 6.2	Phase response of obtained differentiator lying between simpson and trapezoidal response	50
Figure 6.3	Impulse response of obtained differentiator at $a= 0.4$	51
Figure 6.4	Pole zero plot obtained by inverting response of integrator	52
Figure 6.5	Pole zero plot of stabilized differentiator	53
Figure 6.6	Proposed model of digital differentiator	53
Figure 6.7	Response of proposed model of digital differentiator	54
Figure 6.8	Magnitude response of Al-Alaoui operator and NMF	54
Figure 6.9	Phase response of Al-Alaoui operator and NMF	55
Figure 6.10	Impulse response of fractional order operator	55
Figure 6.11	Pole zero plot of Al-Alaoui operator	56
Figure 6.12	Hardware requirement for Integer order digital differentiator	56
Figure 6.13	Hardware requirement for Fractional order digital differentiator	57

LIST OF TABLES

Table 6.1	Magnitude response at normalized frequency at different values of 'a'	50
Table 6.2	Phase response at normalized frequency at different values of 'a'	51

ABBREVIATIONS

ADC	Analog To Digital Converter
DAC	Digital To Analog Converter
FPGA	Field Programmable Gate Array
ASIC	Application Specific Integrated Circuit
DSP	Digital Signal Processing
RF	Radio Frequency
IIR	Infinite Impulse Response
FIR	Finite Impulse Response
SA	Simulated Annealing
DFOD	Digital Fractional Order Differentiator
CFE	Continued Fraction Expansion
FDE	Fractional Order Differential Equation
PSE	Power Series Expansion
SDDD	Second Degree Digital Differentiator
EPLL	Enhanced Phase-Locked-Loop
DFOSGD	Digital Fractional Order Savitzky-Golay Differentiator
DFT	Discrete Fourier Transform
FFT	Fast Fourier Transform
ROC	Region Of Convergence
BIBO	Bounded Input Bounded Output
MATLAB	Matrix Laboratory

CHAPTER 1

INTRODUCTION

1.1 FILTERS:

In signal processing, the function of a filter is to remove unwanted parts of the signal, such as random noise, or to extract useful parts of the signal, such as the components lying within a certain frequency range. Filtering is a class of signal processing, the defining feature of filters being the complete or partial suppression of some aspect of the signal. Most often, this means removing some frequencies and not others in order to suppress interfering signals and reduce background noise. For example, the bass control on a stereo system alters the low-frequency content of a signal, while the treble control alters the high-frequency content. Two common filtering applications are removing noise and decimation. Decimation consists of lowpass filtering and reducing the sample rate [1].

The following block diagram illustrates the basic idea.



Fig 1.1: General filtering method.

1.2 TYPES OF FILTERS

- (a) Analog Filters
- (b) Digital Filters

1.2.1 Analog Filters: An analog filter has an analog signal at both its input $x(t)$ and its output $y(t)$. Both $x(t)$ and $y(t)$ are functions of a continuous variable t and can have an infinite number of values. An analog filter uses analog electronic circuits made up from components such as resistors, capacitors and op-amps to produce the required filtering effect. Such filter circuits are widely used in such applications as noise reduction, video signal enhancement, graphic equalizers in hi-fi systems, and many other areas.

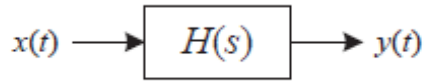


Fig 1.2: Analog implementation of Filter [3].

At all stages, the signal being filtered is an electrical voltage or current which is the direct analogue of the physical quantity (e.g. a sound or video signal or transducer output) involved.

1.2.2 Digital Filters: A digital filter is a system that performs mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal. This is in contrast to the other major type of electronic filter, the analog filter, which is an electronic circuit operating on continuous-time analog signals. An analog signal may be processed by a digital filter by first being digitized and represented as a sequence of numbers, then manipulated mathematically, and then reconstructed as a new analog signal (see digital signal processing) [2].

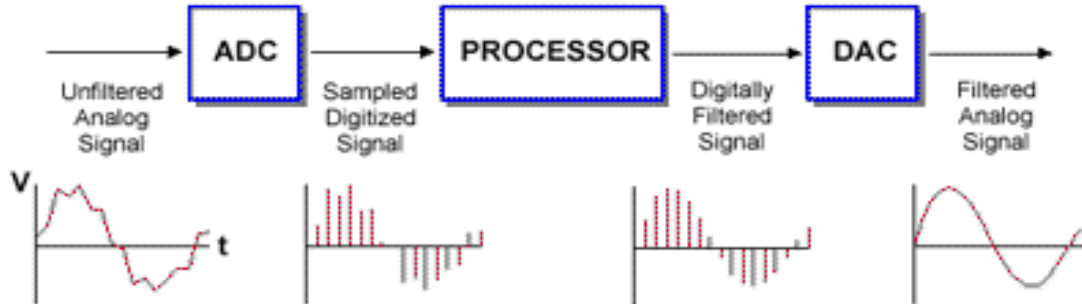


Fig 1.3: Block diagram of Digital Filter [3].

In an analog filter, the input signal is "directly" manipulated by the circuit. A digital filter system usually consists of an analog-to-digital converter to sample the input signal, followed by a microprocessor and some peripheral components such as memory to store data and filter coefficients etc. Finally a digital-to-analog converter to complete the output stage. Program Instructions (software) running on the microprocessor implement the digital filter by performing the necessary mathematical operations on the numbers received from the ADC [4].

In some high performance applications, an FPGA or ASIC is used instead of a general purpose microprocessor, or a specialized DSP with specific paralleled architecture for expediting operations such as filtering [4].

1.3 ADVANTAGES OF USING DIGITAL FILTERS

The following list gives some of the main advantages of digital over analog filters [3].

1. A digital filter is *programmable*, i.e. its operation is determined by a program stored in the processor's memory. This means the digital filter can easily be changed without affecting the circuitry (hardware). An analog filter can only be changed by redesigning the filter circuit.
2. Digital filters are easily *designed, tested and implemented* on a general-purpose computer or workstation.
3. The characteristics of analog filter circuits (particularly those containing active components) are subject to drift and are dependent on temperature. Digital filters do not suffer from these problems, and so are extremely *stable* with respect both to time and temperature.
4. Unlike their analog counterparts, digital filters can handle *low frequency* signals accurately. As the speed of DSP technology continues to increase, digital filters are being applied to high frequency signals in the RF (radio frequency) domain, which in the past was the exclusive preserve of analog technology.
5. Digital filters are very much more *versatile* in their ability to process signals in a variety of ways; this includes the ability of some types of digital filter to adapt to changes in the characteristics of the signal.
6. Fast DSP processors can handle complex combinations of filters in parallel or cascade (series), making the hardware requirements relatively *simple and compact* in comparison with the equivalent analog circuitry.

1.4 APPLICATIONS OF DIGITAL FILTERS

The applications of digital filter are widespread and include but are not limited to the following [5]:

1. Tone Detection (Communications, FAXs, Modems, Biomedical Instrumentation, Acoustical Instrumentation, ATE, etc.): Switched-capacitor filters are almost always the best choice here by virtue of their accurate center frequencies and small board space requirements.
2. Noise Rejection (Line-Frequency Notches for Biomedical Instrumentation and ATE, Low-Pass Noise Filtering for General Instrumentation, Anti-Alias Filtering for Data Acquisition Systems, etc.): All of these applications can be handled well in most cases by either switched-capacitor or conventional active filters. Switched-capacitor filters can run into trouble if the signal bandwidths are high enough relative to the center or cutoff frequencies to cause aliasing, or if the system requires dc precision. Aliasing problems can often be fixed easily with an external resistor and capacitor, but if dc precision is needed, it is usually best to go to a conventional active filter built with precision op amps.
3. Controllable, Variable Frequency Filtering (Spectrum Analysis, Multiple-Function Filters, Software-Controlled Signal Processors, etc.): Switched-capacitor filters excel in applications that require multiple center frequencies because their center frequencies are clock-controlled. Moreover, a single filter can cover a center frequency range of 5 decades. Adjusting the cutoff frequency of a continuous filter is much more difficult and requires analog switches (suitable for a small number of center frequencies), voltage-controlled amplifiers (poor center frequency accuracy) or DACs (good accuracy over a very limited control range).
4. Audio Signal Processing (Tone Controls and Other Equalization, All-Pass Filtering, Active Crossover Networks, etc.): Switched-capacitor filters are usually too noisy for “high-fidelity” audio applications. With a typical dynamic range of about 80 dB to 90 dB, a switched-capacitor filter will usually give 60 dB to 70 dB signal-to-noise ratio (assuming 20 dB of headroom). Also, since audio filters usually need to handle three decades of signal frequencies at the same time, there is a possibility of aliasing problems. Continuous filters are a better choice for general audio use, although many communications systems have bandwidths and S/N ratios that are compatible with switched capacitor filters, and these systems can take advantage of the tunability and small size of monolithic filters.

1.5 ORGANIZATION OF THE DISSERTATION

This dissertation is organized as follows

Chapter 2 contains the literature review of papers related to the work. It contains the literature review of papers used in the designing of different types of IIR filters.

Chapter 3 focuses on various characteristics of digital filters. It also looks into response of filters in time domain and in frequency domain while showing the response between poor and good digital filters.

Chapter 4 discusses on design methodologies for IIR digital filter design. It also looks into various types of IIR filters and comparing their responses. It also gives the algorithm for appropriate digital filter design.

Chapter 5 discusses on the mathematical design of the IIR digital differentiator using various transformation techniques. It also gives insight on the technique for stabilizing poles lying outside the unit circle.

Chapter 6 shows the response obtained for the various mathematical model of digital differentiator which were developed in previous sections. Magnitude, frequency and pole zero plot for all the designed differentiators have been shown.

Chapter 7 contains conclusion and future scope of the work.

CHAPTER 2

LITERATURE REVIEW

M. A. Al-Alaoui [6] presented a novel class of stable, minimum phase, second-order, lowpass IIR digital differentiator. It is obtained by inverting the transfer functions of a class of second-order integrators, stabilizing the resulting transfer functions, and compensating their magnitudes. The class of second-order integrator is obtained by interpolating the traditional Simpson and trapezoidal integrators. The resulting integrators have a perfect -90° phase over the Nyquist interval.

In addition to the above two integrators, the Tick integrator is also a member of the class. The resulting integrators and differentiators extend the frequency range of operation beyond that possible by using either of the two traditional integrators. The resulting integrators better approximate the magnitude characteristics of the ideal integrator by widening the frequency range where the desired response is captured as compared to the two traditional integrators. The integrators integrate constant and linear inputs exactly and exhibit perfect phase over the entire Nyquist interval. Additionally there is one free parameter that can be adjusted by imposing an appropriate constraint.

The low-pass differentiators have the desirable magnitude response of zero at zero frequency and at the Nyquist frequency. A set of minimax differentiators, and integrators, is obtained by minimizing the maximum magnitude error in the lowest Xth fraction of the Nyquist interval. The phases of the resulting differentiators are almost linear over the frequency ranges of interest. The low orders and high accuracies of the integrators and differentiators make them attractive for real time applications.

M. A. Al-Alaoui [7] discussed a novel approach to designing recursive stable digital differentiators. A four-step design procedure is presented. The procedure consists of obtaining or designing an integrator and then modifying its transfer function appropriately to obtain a stable differentiator. As an example a second order recursive differentiator is developed.

In this letter, as an example, a new differentiator is obtained by inverting the transfer functions of the Tick integrator. The procedure had been tested successfully on Simpson and other integrators. Tick designed a transfer function that is as close to unity as possible throughout the lower half of the Nyquist interval while still involving only three consecutive terms. In a sense, the Tick integrator is an optimised simpson integrator. The low order of the differentiator makes it suitable for real-time applications.

B.K. Lau et al. [8] presented a new method for the design of approximately linear phase IIR filters is proposed. This technique is based on first designing an approximately linear phase asymmetrical FIR filter which satisfies the magnitude specifications and then obtaining the low order IIR filter. The design of an asymmetrical FIR filter begins with the design of a symmetrical FIR filter satisfying the given specifications.

Using the symmetrical filter, the asymmetrical filter is obtained with the shift, truncate, and zero pad technique. The shifting process involves shifting the impulse response coefficients of the filter to the left so that the filter becomes anticausal. The truncation process involves truncating the anticausal part of the filter. The zero pad process involves padding the filter to the right by zeros so that the order of the filter is restored to the original order. This filter is then reduced using model reduction techniques to obtain a low order IIR filter which meets the original magnitude response specifications while maintaining an approximately linear phase characteristic in the passband.

This technique gives better results than the equalization approach in terms of the filter order. Numerical studies indicate that the order of the IIR filters obtained using asymmetrical FIR filters (as proposed in the paper) is considerably lower than those obtained using symmetrical FIR filters.

I. W. Selesnick [9] proposed a paper which describes the design of type III and type IV linear-phase finite-impulse response (FIR) low-pass digital differentiators according to the maximally flat criterion. The paper introduces a two-term recursive formula that enables the simple stable computation of the impulse response coefficients. The same recursive formula is valid for both Type III and Type IV solutions. The solutions cannot be obtained from a low-pass filter as in the case of a full-band differentiator.

The algorithms for automatic sum simplification are used in this paper to obtain a simple two-term recurrence relation for computing the coefficients of the impulse response. There are several possible extensions to the problem described in this paper. For example, the extension of the recursive formulas to the case where the maximally flat approximation to the ideal differentiator is performed not at $\omega = 0$ but at another frequency ω_0 .

This type of solution is relevant when the signal is centred on a known frequency. Another remaining question is the existence of low-complexity structures for maximally flat differentiators. Those structures are multiplier less and have a regular structure. Another extension of the approach described in this paper is the design of second (and higher) order differentiators where the desired frequency response is $-\omega^2$, etc.

M. A. Al-Alaoui [10] proposed a new class of digital IIR integrators and a corresponding class of digital IIR differentiators. The transfer functions of the new integrators were obtained by applying the z-transform to the different numerical integration rules that resulted from interpolating Newton-Cotes integration rules. The transfer functions of the resulting integrators were inverted and stabilized to obtain the corresponding digital differentiators. The resulting integrators and differentiators are viable alternatives to available integrators and differentiators.

In some cases the obtained transfer functions have poles that lie on the unit circle of the z-plane which obviates the application of the stabilizing method which consists of reflecting the pole that lies outside the unit circle at radius r to inside the unit circle at a radius of $1/r$ and compensating the magnitude response by multiplying the resulting transfer function by $1/r$. To overcome the challenge presented by the poles that lie on the unit circle, the optimizing Simulated Annealing algorithm was applied.

Improved results were manifested by eliminating the spurious responses due to the unit circle poles, improving the magnitude and phase responses and outperforming the best wideband differentiator. The convergence of the SA method depends on the initial conditions. One may argue that the simulated annealing method is a random search algorithm, so the mean and standard deviation of the design results should be reported and that it is meaningless to give the best result as was done above.

Integrators and differentiators derived from specific integration rules then optimized using SA to primarily eliminate offending poles. So, in this case, only the final result is required. Here the performance of the obtained integrators and differentiators is studied, not of SA. In the computational intelligence area, there are genetic algorithm, simulated annealing (SA) algorithm, differential evolution algorithm, particle swarm algorithm, colony optimization algorithm, and Tabu search algorithm etc. The objective of the paper is not to argue the superiority of SA over other algorithms but to study the performance of obtained integrators and differentiators.

The contribution of the work is the derivation of new integrators and differentiators from numerical integration rules. When offending poles were encountered during this derivation, SA was just a proposed mean to eliminate them. SA is not the main issue in the paper. Using another algorithm could yield comparable results. But comparing the different optimization algorithms is not the aim of this paper. The paper proposed new efficient integrators and differentiators.

M. A. Al-Alaoui [11] presented a paper in which Novel stable higher order s -to- z transforms are introduced. A class of second-order stable transforms is obtained by stabilizing the class of second-order integrators resulting from interpolating the Simpson and the trapezoidal rules. In addition new transforms are obtained by stabilizing the s -to- z mapping functions obtained from the Adams–Moulton numerical integration formulas. The approach is general and can be applied to other integration rules to obtain stable transforms.

An important advantage of the new stabilized rules is that they yield minimum phase filters when applied to analog all-pole filters. The transforms were obtained by stabilizing the unstable mappings obtained from the numerical integration rules. The examples demonstrate the viability of the proposed transforms. The accuracy of the new transforms are derived from the accuracy of the integration rules from which they are obtained. An important advantage of the new stabilized rules is that they yield minimum-phase filters when applied to analog all-pole filters.

A. Loverro [12] proposed a paper aimed at the engineering and scientific professionals who wish to learn about Fractional Calculus and its possible applications in their field of study.

The intent is to first expose the reader to the concepts, applicable definitions, and execution of fractional calculus and second to show how these may be used to solve several modern problems. Only the very basic concepts regarding fractional order calculus were addressed here. The paper tells about the gaps of traditional calculus in ways that as of yet, no one completely understands. But the goal of this paper is not only to expose the reader to the basic concepts of fractional calculus, but also to whet their appetite with scientific and engineering applications.

M. A. Al-Alaoui [13] presented a novel approach to design approximately linear phase infinite-impulse-response (IIR) digital filters in the passband region. The proposed approach yields digital IIR filters whose numerators represent linear phase finite-impulse-response (FIR) filters. As an example, low-pass IIR differentiators are introduced. The range and high-frequency suppression of the proposed low-pass differentiators are comparable to those obtained by higher order FIR low-pass differentiators.

In addition, the differentiators exhibit almost linear phases in the passband regions. Two novel approaches to designing linear phase recursive digital low-pass differentiators were presented. The approaches utilize the linear phase properties of the FIR filters and the steeper magnitude roll-off properties of the IIR filters to obtain IIR low-pass digital differentiators. The proposed low-pass differentiators were shown to have shorter transition regions, and thus better ability to suppress high frequency noise, for much lower order filters, than the corresponding FIR filters.

In addition, the new low-pass differentiators exhibit almost linear phases in their corresponding passband. The examples demonstrate the versatility of the proposed approaches. The low order of the resulting differentiators makes them suitable for real-time applications. The new low-pass differentiators compare favorably with the state-of-the-art low-pass FIR digital differentiators. In addition, the approaches could well be employed to design other types of IIR filters that approximate linear phases in the passbands and meet the magnitude specifications at a lower computational cost than the corresponding FIR filters.

Y. Q. Chen et al. [14] presented a new infinite impulse response (IIR)-type digital fractional order differentiator (DFOD) by using a new family of first-order digital differentiators

expressed in the second-order IIR filter form. The integer first-order digital differentiators are obtained by the stable inversion of the weighted sum of Simpson integration rule and the trapezoidal integration rule. After that the corresponding fractional-order digital differentiator via CFE (Continued Fraction Expansion) truncation is presented. The distinguishing point of the proposed DFOD lies in an additional tuning knob to compromise the high-frequency approximation accuracy.

Y. Q. Chen et al. [15] presented a paper for fractional-order differentiator s^r where r is a real number, describing its discretization step for its digital implementation. Two discretization methods are presented. The first scheme is a direct recursive discretization of the Tustin operator. The second one is a direct discretization method using the Al-Alaoui operator via continued fraction expansion (CFE). This brief firstly focuses on the direct discretization method using the well-known Tustin operator which is a straightforward scheme to discretize the fractional-order derivative.

The major contribution of this brief is to introduce a recursive formula for discretization with different order of approximation which simplifies the programming efforts. Moreover, the discretized transfer function is stable and minimum phase. However, the Tustin operator based discretization scheme exhibits large errors in high frequency range. A new mixed scheme of Euler and Tustin operators is proposed which yields the Al-Alaoui operator.

Using the continued fraction expansion of the Al-Alaoui operator, this paper contributes a new direct discretization scheme with a very good magnitude fit to that of the original continuous fractional differentiator. The approximate discretization is minimum phase and stable. Detailed discretization procedures and short MATLAB scripts are given. Examples are included for illustration.

L. Dorcak et al. [16] presented some alternative types of discretization methods (discrete approximation) for the fractional-order (FO) differentiator and their application to the FO dynamical system described by the FO differential equation (FDE). With analytical solution and numerical solution by power series expansion (PSE) method are compared two effective methods - the Muir expansion of the Tustin operator and continued fraction expansion

method (CFE) with the Tustin operator and the Al-Alaoui operator. Except detailed mathematical description presented are also simulation results.

From the Bode plots of the FO differentiator and FDE and from the solution in the time domain it is seen, that the CFE is a more effective method according to the PSE method, but there are some restrictions for the choice of the time step. The Muir expansion is almost unusable.

C. C. Tseng [17] investigated the design of fractional order Simpson digital integrator. First, the conventional transfer function of second-order IIR Simpson integrator is factorized into product of first-order factors. Then, each factor of product is fractionalized by taking fractional power and using binomial series expansion. Next, truncating the length of series expansion, the transfer functions of FIR and IIR fractional order Simpson integrators are obtained. It is shown that the IIR fractional integrator is always stable. Finally, design examples are demonstrated to illustrate the performance of the proposed method and some further improvement using fractional delay is also discussed.

M. A. Al-Alaoui [18] deals with the simulation and discretization of continuous time models with fractional order differentiation or integration. It addresses the issue of direct discretization of analog fractional order systems followed by approximating the discretization result with an integer order transfer function or approximating the analog fractional order system with an integer analog order system and then discretizing the resulting integer analog approximation. It provides a counterexample to the claim that analog integer order approximation followed by discretization is superior to the direct discretization approach.

B. T. Krishna et al. [19] presented a paper which deals with the design and applications of digital differentiators. For real time applications it is mandatory that a differentiator should have as small an order as possible. Different procedures available for the design of FIR and IIR type digital differentiators are presented. The IIR Type digital differentiators are obtained by inversion and magnitude stabilization of the existing digital integrators. In some applications like controls, wave shaping, oscillators and communications require a constant 90° phase for differentiators.

In this paper an attempt is made to study about the variation of phase angle of digital differentiators with the application of fractional delay. The use of digital differentiators for the detection of edges in an image, QRS detection in an ECG signal is illustrated. It has been observed that the digital differentiators have shown superior performance compared to the well-known gradient method. With the proven efficiency of the differentiators in various applications, they have been implemented in hardware using Verilog. A study on the design of digital differentiators reveals that type III FIR differentiators have the inherent nature in frequency response of approaching zero at Nyquist frequency.

To extend the performance of type III FIR filters in the higher frequency bands, one has to increase the filter taps. Type IV FIR differentiators using Fourier series have been found to have outstanding frequency response, however, they are noisy and biased. Higher order central difference approximations using Taylor series might outperform windowed Fourier series since there is no truncation and the associated Gibbs phenomenon. IIR digital differentiators are more favourable for real-time application. The first-order IIR differentiator from Al-Alaoui is ideal in terms of the frequency response. The IIR differentiators can be adaptively used when systems experience high dynamics. The digital differentiators of IIR type have been proved to be much more efficient in detecting edges of an image, QRS detection etc.

D.K. Upadhyay et al. [20] modelled a new design of a recursive wideband digital differentiator which is obtained by optimising the pole-zero locations of existing recursive wideband digital differentiators. Further, a new integrator design is also obtained by inverting the transfer function of the proposed digital differentiator design with suitable modifications. The beauty of these designs is that they are only of second-order recursive systems and have not more than 0.48% relative error in magnitude responses almost over the full Nyquist band, thus being suitable for real-time circuit applications.

Results show that the proposed designs have not more than 0.48% relative error in magnitude responses over the full Nyquist band except near $\omega=\pi$ and have either improved or comparable phase responses with existing designs of higher-order recursive systems. Thus, attractive realtime signal processing applications can be achieved using the proposed differentiator and integrator designs, which are of second-order recursive systems.

A. Sarkar et al. [21] presented a study highlighting development and implementation of a novel digital signal processing algorithm for on-line estimation of the fundamental frequency of non-sinusoidal power system signals. The basic algorithm relies on the development of a new variance reduction algorithm and design of an optimised infinite impulse response (IIR) second-degree digital differentiator (SDDD).

The design of SDDD consists of obtaining a second degree integrator from Tick integrator, and then modifying and optimising its transfer function appropriately to attain a stable second-degree differentiator. Compared with the well-established technique such as the enhanced phase-locked-loop (EPLL) system, the proposed algorithm provides faster transient response, higher degree of immunity and insensitivity to harmonics and noise. Structural simplicity, wide range of application and robustness against sampling frequency variation are other salient features of the method. The only limitation as compared with the EPLL system is its slight reduced accuracy (around 3 mHz) under static sinusoidal conditions.

Based on simulation studies, performances of the proposed algorithm at different operating conditions have been presented and its accuracy and response time have been compared with the EPLL systems. Apart from power system applications, the proposed algorithm can also be utilised as a reference for sampling control in analog-to-digital conversion during power quality disturbance measurement and for synchronisation of power electronic converters in polluted and variable-frequency environments. Structural simplicity of the proposed estimator makes it suitable for digital implementation in both software environment, e.g., a digital signal processor (DSP), and a digital hardware environment, e.g., field programmable gate array (FPGA) or application-specific integrated circuit (ASIC).

D. Chen et al. [22] presented a brief study on design method for a digital fractional order Savitzky-Golay differentiator (DFOSGD), which generalizes the Savitzky-Golay filter from the integer order to the fractional order for estimating the fractional order derivative of the contaminated signal. The proposed method calculates the moving window's weights using the polynomial least-squares method and the Riemann-Liouville fractional order derivative definition, and then computes the fractional order derivative of the given signal using the

convolution between the weights and the signal, instead of the complex mathematical deduction.

Hence, the computation time is greatly improved. Frequency-domain analysis reveals that the proposed differentiator is essentially a fractional order low-pass filter. Experiments demonstrate that the proposed DFOSGD can accurately estimate the fractional order derivatives of both noise-free signal and contaminated signal. Four experiments were performed to evaluate the DFOSGD, and the results demonstrated its validity.

The proposed DFOSGD includes the following advantages; First, it is the first time to combine the fractional calculus with the Savitzky-Golay filter and provides a new way for designing the DFOD. Second, it can accurately and easily estimate the fractional order derivative of the contaminated signal. Future works involve applying the proposed DFOSGD in image enhancement.

M. A. Al-Alaoui [23] discussed new transformation polynomials for discretization of analogue systems. The work proposes that the discretization of $1/s^n$ should be done independently rather than by raising the discrete representation of $1/s$ to the power n . Several examples are given in to back this idea. This paper shows that the new transformation polynomials for discretization of analogue systems are exactly the same as the parameterized Al-Alaoui operator. Al-Alaoui operator is a credible alternative to other discretization methods with wide range of applications in integer and fractional order discretization and other applications that require accurate and real time differentiation or integration.

F. Leulmi et al. [24] proposed an improvement of the rational approximation of continuous fractional order differentiators and integrators of type s^a obtained using direct discretization approach of power series expansion-signal modelling technique by using a new second order s-to-z transform obtained by inversion and stabilizing the digital integrator designed from conventional Simpson integrator and fractional delay filter. The obtained new IIR type digital fractional order differentiators have much better frequency characteristic as compared to the Al-Alaoui operator based approximations.

CHAPTER 3

DIGITAL FILTERS

3.1 OVERVIEW

Digital filters are a very important part of DSP. In fact, their extraordinary performance is one of the key reasons that DSP has become so popular. As mentioned in the introduction, filters have two uses: signal *separation* and signal *restoration*. Signal separation is needed when a signal has been contaminated with interference, noise, or other signals[25].

Signal restoration is used when a signal has been distorted in some way. For example, an audio recording made with poor equipment may be filtered to better represent the sound as it actually occurred. Another example is the deblurring of an image acquired with an improperly focused lens, or a shaky camera.

A digital filter is a basic building block in any Digital Signal Processing (DSP) system. The frequency response of the filter depends on the value of its coefficients, or taps. Many software programs can compute the values of the coefficients based on the desired frequency response. These values are typically floating point numbers and they are represented with a fairly high degree of precision. However, when a digital filter is implemented in hardware, the designer wants to represent the coefficients (and also the data) with the smallest number of bits that still gives acceptable resolution for the numbers.

This is because representing a number with excess bits increases the size of the registers, buses, adders, multipliers and other hardware used to process that signal. The bigger sizes result in a chip with a larger die size, which translates into increased power consumption and a higher chip price. Thus, the bit precisions used to represent numbers are important in the performance of real-world signal processing designs[25].

It is common in DSP to say that a filter's input and output signals are in the time domain. This is because signals are usually created by sampling at regular intervals of time. But this is not the only way sampling can take place. The second most common way of sampling is at equal intervals in space. Many other domains are possible; however, time and space are by far the most common. When we refer the term time domain in DSP, it actually

refers to samples taken over time, or it may be a general reference to any domain that the samples are taken in.

As shown in Fig. 3.1, every linear filter has an impulse response, a step response and a frequency response. Each of these responses contains complete information about the filter, but in a different form. If one of the three is specified, the other two are fixed and can be directly calculated. All three of these representations are important, because they describe how the filter will react under different circumstances.

The most straightforward way to implement a digital filter is by convolving the input signal with the digital filter's impulse response. All possible linear filters can be made in this manner. When the impulse response is used in this way, filter designers give it a special name: the filter kernel.

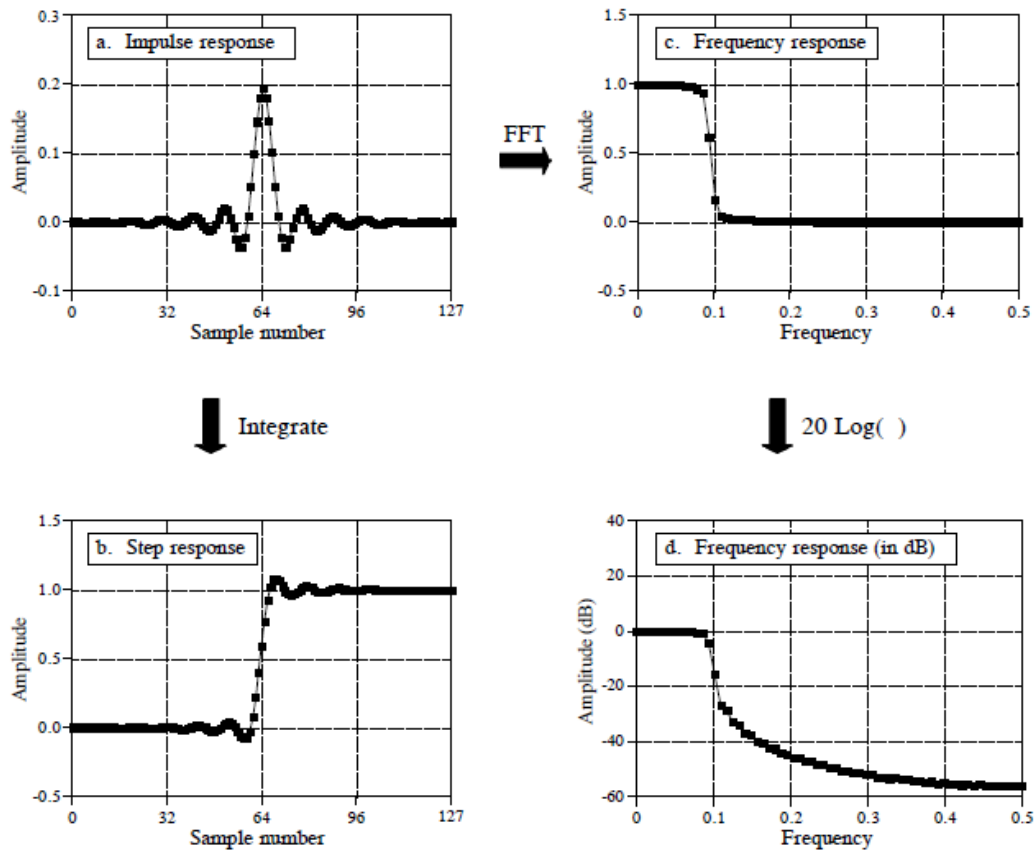


Fig 3.1: Filter parameters. (a) Impulse response (b) step response (c) frequency response on linear scale & (d) decibel scale [25].

There is also another way to make digital filters, called recursion. When a filter is implemented by convolution, each sample in the output is calculated by weighting the

samples in the input, and adding them together. Recursive filters are an extension of this, using previously calculated values from the output, besides points from the input. Instead of using a filter kernel, recursive filters are defined by a set of recursion coefficients.

The impulse responses of recursive filters are composed of sinusoids that exponentially decay in amplitude. In principle, this makes their impulse responses infinitely long. However, the amplitude eventually drops below the round-off noise of the system, and the remaining samples can be ignored. Because of this characteristic, recursive filters are also called Infinite Impulse Response or IIR filters. In comparison, filters carried out by convolution are called Finite Impulse Response or FIR filters [25].

3.2 IMPULSE RESPONSE: The impulse response is the output of a system when the input is an impulse. Any LTI filter may be represented in the time domain by its response to a specific signal called the impulse. This response is called, naturally enough, the impulse response of the filter. Any LTI filter can be implemented by convolving the input signal with the filter impulse response, as we will see [26].

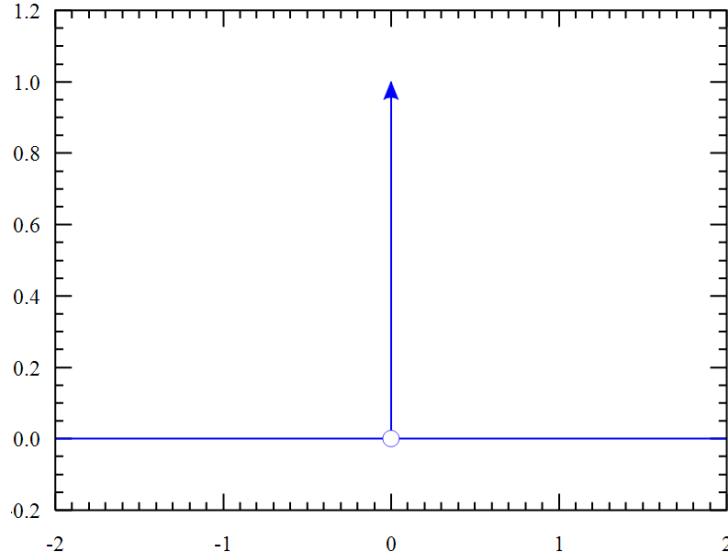


Fig 3.2: Impulse Response.

The impulse signal is denoted $\delta(n)$ and defined by:

$$\delta(n) = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases}$$

The impulse response of a filter is the response of the filter to $\delta(n)$ and is most often denoted $h(n)$:

$$h(n) \triangleq \mathcal{L}_n\{\delta(\cdot)\}$$

The impulse response $h(n)$ is the response of the filter \mathcal{L} at time n to a unit impulse occurring at any instant of time.

3.3 FILTER CHARACTERISTICS:

3.3.1 HOW INFORMATION IS REPRESENTED IN SIGNALS

The most important part of any DSP task understands how information is contained in the signals you are working with. There are many ways that information can be contained in a signal. This is especially true if the signal is manmade. For instance, consider all of the modulation schemes that have been devised: AM, FM, single-sideband, pulse-code modulation, pulse-width modulation, etc. The list goes on and on. Fortunately, there are only two ways that are common for information to be represented in naturally occurring signals. We will call these: information represented in the time domain, and information represented in the frequency domain [25].

Information represented in the time domain describes when something occurs and what the amplitude of the occurrence is. For example, imagine an experiment to study the light output from the sun. The light output is measured and recorded once each second. Each sample in the signal indicates what is happening at that instant, and the level of the event. If a solar flare occurs, the signal directly provides information on the time it occurred, the duration, the development over time, etc. Each sample contains information that is interpretable without reference to any other sample. Even if you have only one sample from this signal, you still know something about what you are measuring. This is the simplest way for information to be contained in a signal.

In contrast, information represented in the frequency domain is more indirect. Many things in our universe show periodic motion. For example, a wine glass struck with a fingernail will vibrate, producing a ringing sound; the pendulum of a grandfather clock swings back and forth; stars and planets rotate on their axis and revolve around each other,

and so forth. By measuring the frequency, phase, and amplitude of this periodic motion, information can often be obtained about the system producing the motion. Suppose we sample the sound produced by the ringing wine glass. The fundamental frequency and harmonics of the periodic vibration relate to the mass and elasticity of the material. A single sample, in itself, contains no information about the periodic motion, and therefore no information about the wine glass. The information is contained in the relationship between many points in the signal.

This brings us to the importance of the step and frequency responses. The step response describes how information represented in the time domain is being modified by the system. In contrast, the frequency response shows how information represented in the frequency domain is being changed. This distinction is absolutely critical in filter design because it is not possible to optimize a filter for both applications. Good performance in the time domain results in poor performance in the frequency domain, and vice versa. If you are designing a filter to remove noise from an EKG signal (information represented in the time domain), the step response is the important parameter, and the frequency response is of little concern. Now let's look at what makes a filter optimal for time domain or frequency domain applications [25].

3.3.2 TIME DOMAIN PARAMETERS

It may not be obvious why the step response is of such concern in time domain filters. You may be wondering why the impulse response isn't the important parameter. The answer lies in the way that the human mind understands and processes information. Remember that the step, impulse and frequency responses all contain identical information, just in different arrangements. The step response is useful in time domain analysis because it matches the way humans view the information contained in the signals.

For example, suppose you are given a signal of some unknown origin and asked to analyze it. The first thing you will do is divide the signal into regions of similar characteristics. You can't stop from doing this; your mind will do it automatically. Some of the regions may be smooth; others may have large amplitude peaks; others may be noisy. This segmentation is accomplished by identifying the points that separate the regions. This is where the step function comes in. The step function is the purest way of representing a

division between two dissimilar regions. It can mark when an event starts, or when an event ends. It tells you that whatever is on the left is somehow different from whatever is on the right. This is how the human mind views time domain information: a group of step functions dividing the information into regions of similar characteristics.

The step response, in turn, is important because it describes how the dividing lines are being modified by the filter. The step response parameters that are important in filter design are shown in Fig. 3.3. To distinguish events in a signal, the duration of the step response must be shorter than the spacing of the events. This dictates that the step response should be as fast as possible. This is shown in Figs. (a) & (b). The most common way to specify the rise time is to quote the number of samples between the 10% and 90% amplitude levels.

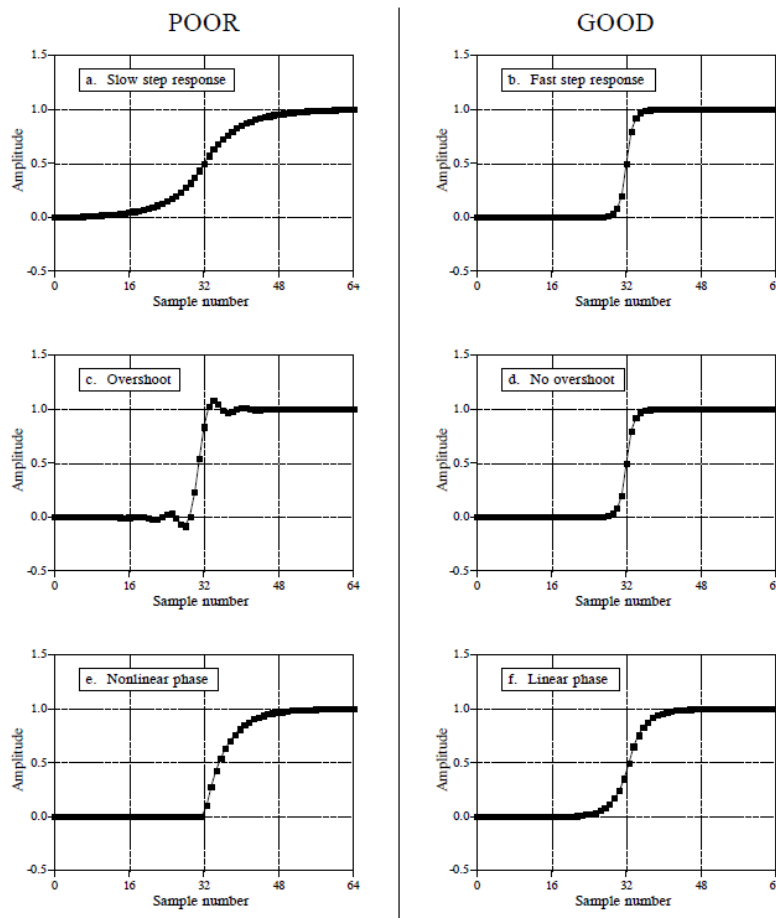


Fig. 3.3: Transition time shown in (a) and (b), overshoot shown in (c) and (d) and phase linearity shown in (e) and (f) [25].

Figures (c) and (d) show the next parameter that is important: overshoot in the step response. Overshoot must generally be eliminated because it changes the amplitude of samples in the signal; this is a basic distortion of the information contained in the time domain. This can be summed up in one question: Is the overshoot you observe in a signal coming from the thing you are trying to measure, or from the filter you have used?

Finally, it is often desired that the upper half of the step response be symmetrical with the lower half, as illustrated in (e) and (f). This symmetry is needed to make the rising edges look the same as the falling edges. This symmetry is called linear phase, because the frequency response has a phase that is a straight line. Make sure you understand these three parameters; they are the key to evaluating time domain filters.

3.3.3 FREQUENCY DOMAIN PARAMETERS

Figure 3.4 shows the four basic frequency responses. The purpose of these filters is to allow some frequencies to pass unaltered, while completely blocking other frequencies. The passband refers to those frequencies that are passed, while the stopband contains those frequencies that are blocked. The transition band is between. A fast roll-off means that the transition band is very narrow. The division between the passband and transition band is called the cutoff frequency.

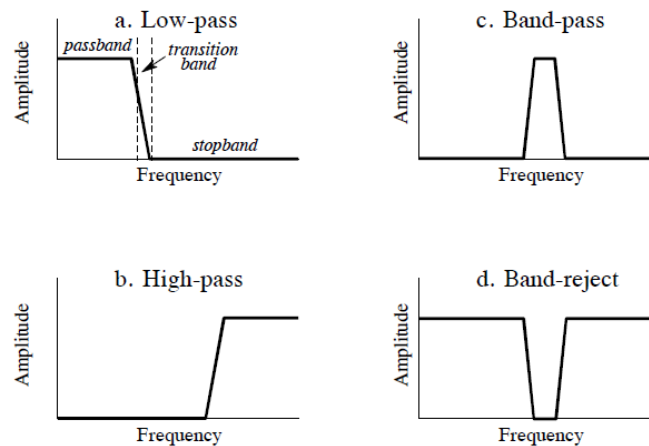


Fig. 3.4: The four common frequency responses of filters: low pass, high pass, band pass and band reject [25].

Figure 3.5 shows three parameters that measure how well a filter performs in the frequency domain. To separate closely spaced frequencies, the filter must have a fast roll-off, as illustrated in (a) and (b). For the passband frequencies to move through the filter unaltered there must be no passband ripple, as shown in (c) and (d). Lastly, to adequately block the stopband frequencies, it is necessary to have good stopband attenuation, displayed in (e) and (f).

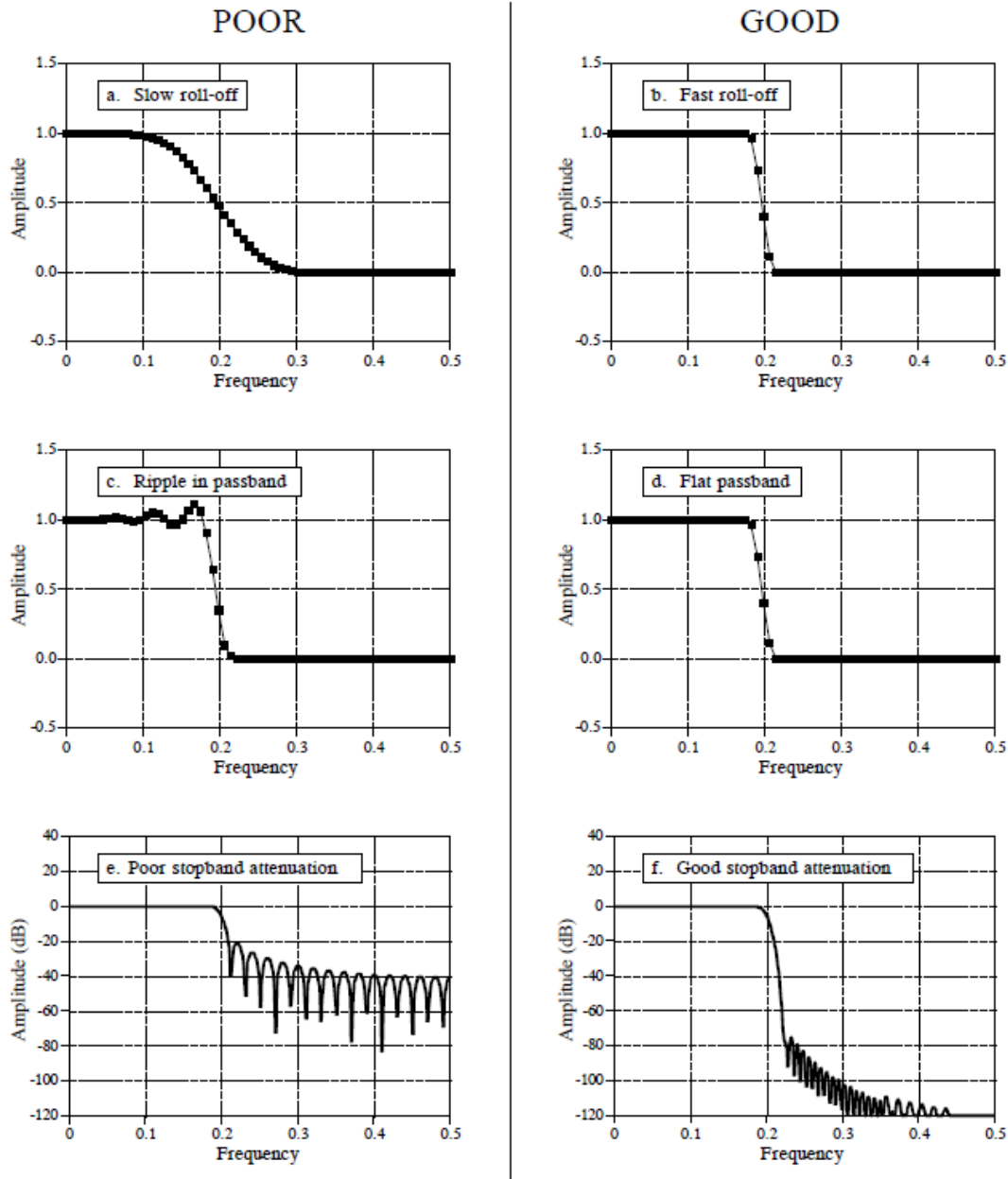


Fig. 3.5: Parameters for evaluating frequency domain parameters: roll off sharpness shown in (a) and (b), passband ripple shown in (c) and (d), stopband attenuation shown in (e) and (f) [25].

Why is there nothing about the phase in these parameters? First, the phase isn't important in most frequency domain applications. For example, the phase of an audio signal is almost completely random, and contains little useful information. Second, if the phase is important, it is very easy to make digital filters with a perfect phase response, i.e., all frequencies pass through the filter with a zero phase shift.

Previous chapters have described how the DFT converts a system's impulse response into its frequency response. The quickest way to calculate the DFT is by means of the FFT algorithm. Starting with filter kernel N samples long, the FFT calculates the frequency spectrum consisting of an N point real part and an N point imaginary part. Only samples 0 to $N/2$ of the FFT's real and imaginary parts contain useful information; the remaining points are duplicates (negative frequencies) and can be ignored.

Since the real and imaginary parts are difficult for humans to understand, they are usually converted into polar notation. This provides the magnitude and phase signals, each running from sample 0 to sample $N/2$ (i.e., $N/2 + 1$ samples in each signal). For example, an impulse response of 256 points will result in a frequency response running from point 0 to 128. Sample 0 represents DC, i.e., zero frequency. Sample 128 represents one-half of the sampling rate. No frequencies higher than one-half of the sampling rate can appear in sampled data.

The number of samples used to represent the impulse response can be arbitrarily large. For instance, suppose you want to find the frequency response of a filter kernel that consists of 80 points. Since the FFT only works with signals that are a power of two, you need to add 48 zeros to the signal to bring it to a length of 128 samples. This padding with zeros does not change the impulse response. To understand why this is so, think about what happens to these added zeros when the input signal is convolved with the system's impulse response.

The added zeros simply vanish in the convolution, and do not affect the outcome. Taking this a step further, you could add many zeros to the impulse response to make it, say, 256, 512, or 1024 points long. The important idea is that longer impulse responses result in a closer spacing of the data points in the frequency response. That is, there are more samples spread between DC and one-half of the sampling rate. Taking this to the extreme, if the

impulse response is padded with an infinite number of zeros, the data points in the frequency response are infinitesimally close together, i.e., a continuous line.

In other words, the frequency response of a filter is really a continuous signal between DC and one-half of the sampling rate. The output of the DFT is a sampling of this continuous line. What length of impulse response should you use when calculating a filter's frequency response? As a first thought, try, but don't be afraid to change it if needed (such as insufficient resolution or excessive computation time).

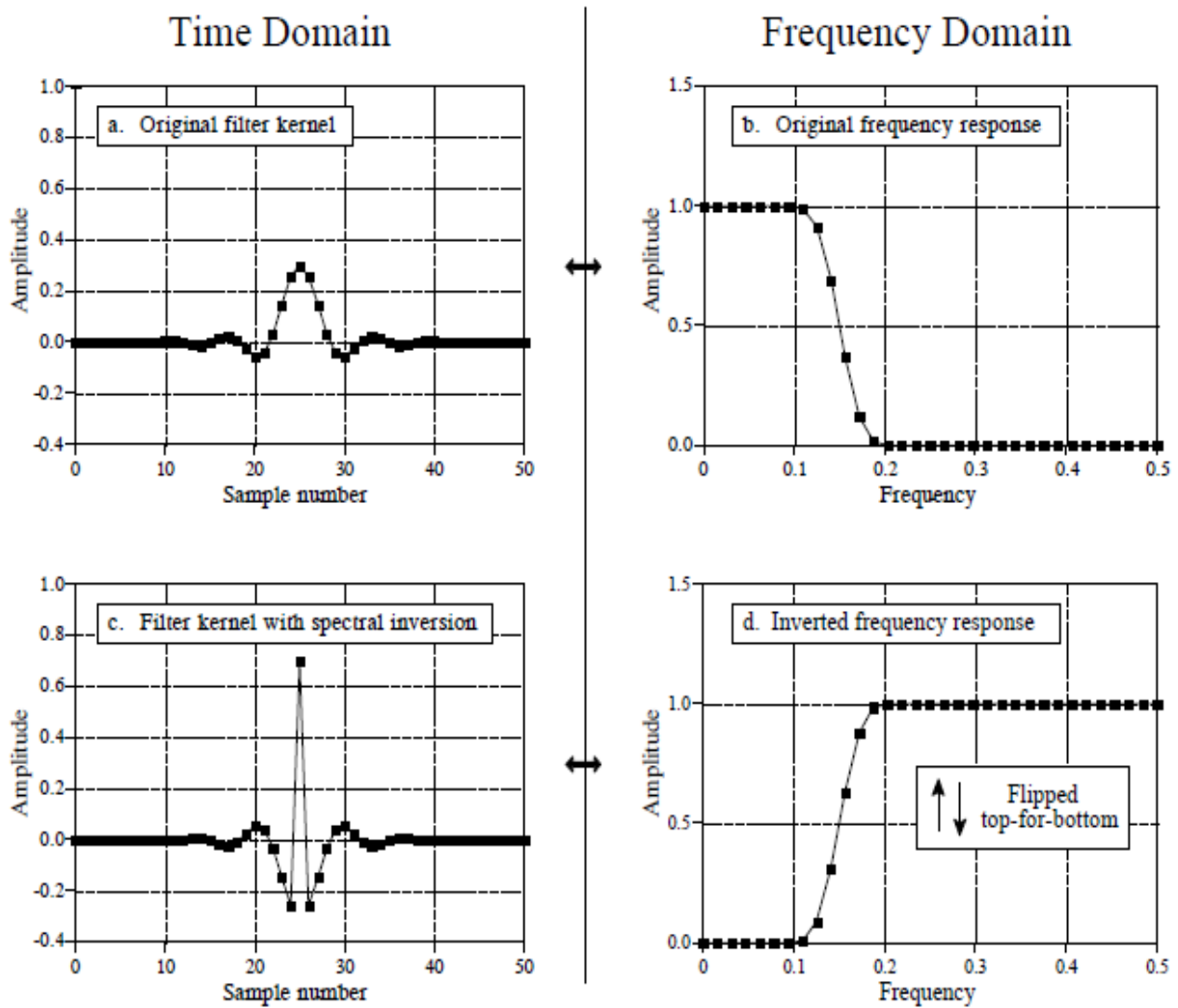


Fig. 3.6: (a) showing impulse response of low pass filter and (b) showing its respective frequency response (c) showing impulse response of high pass filter and (d) showing its respective frequency response [25].

3.4 TYPES OF DIGITAL FILTERS

3.4.1 IIR FILTERS

IIR Filters are systems which are distinguished by having an impulse response which does not become exactly zero past a certain point, but continues indefinitely. The presence of feedback in the topology of a discrete-time filter generally creates an IIR response.

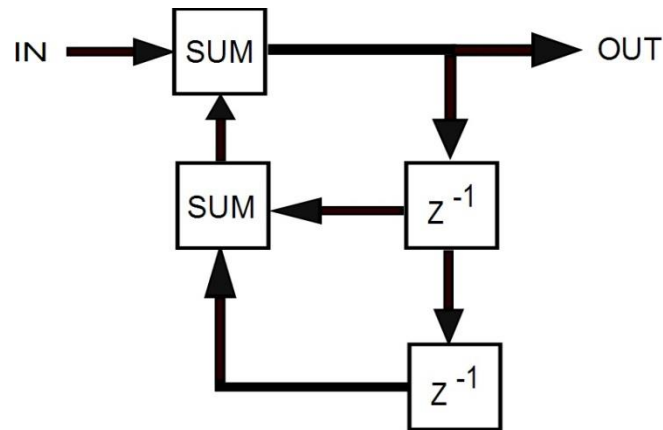


Fig. 3.7: General IIR filter structure.

The transfer function of IIR Filter is defined as

$$H(z) = \frac{Y(z)}{X(z)}$$
$$H(z) = \frac{\sum_{i=0}^P b_i z^{-i}}{1 + \sum_{j=0}^Q a_j z^{-j}}$$

where P is feedforward filter order, b_i are the feedforward filter coefficients, Q is feedback filter order, a_j are the feedback filter coefficients. Digital IIR filters have been derived from their analog counterparts. The digital version of these can be obtained from analog designs through the bilinear transformation. In short, the bilinear transformation is an algebraic mapping from the continuous frequency variable s to the discrete frequency variable z such that the imaginary axis in the s -plane corresponds to one revolution of the unit circle in the z -plane [27]:

$$s = \frac{1 - z^{-1}}{1 + z^{-1}}$$

Advantages & Disadvantages of IIR Filter:

The main advantage digital IIR filters have over FIR filters is their efficiency in implementation, in order to meet a specification in terms of passband, stopband, ripple, and/or roll-off. Such a set of specifications can be accomplished with a lower order (Q in the above formulae) IIR filter than would be required for an FIR filter meeting the same requirements. If implemented in a signal processor, this implies a correspondingly fewer number of calculations per time step; the computational savings is often of a rather large factor [27].

On the other hand, FIR filters can be easier to design, for instance, to match a particular frequency response requirement. This is particularly true when the requirement is not one of the usual cases (high-pass, low-pass, notch, etc.) which have been studied and optimized for analog filters. Also FIR filters can be easily made to be linear phase (constant group delay vs frequency), a property that is not easily met using IIR filters and then only as an approximation (for instance with the Bessel filter). Another issue regarding digital IIR filters is the potential for limit cycle behavior when idle, due to the feedback system in conjunction with quantization.

The main difference between IIR filters and FIR filters is that an IIR filter is more compact in that it can usually achieve a prescribed frequency response with a smaller number of coefficients than an FIR filter. A smaller number of filter coefficients imply less storage requirements and faster calculation and a higher throughput. Therefore, generally IIR filters are more efficient in memory and computational requirements than FIR filters. However, it must be noted that an FIR filter is always stable, whereas an IIR filter can become unstable (for example if the poles of the IIR filter are outside the unit circle) and care must be taken in design of IIR filters to ensure stability.

3.4.2 FIR FILTERS

In signal processing, a finite impulse response (FIR) filter is a filter whose impulse response (or response to any finite length input) is of *finite* duration, because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which may have internal feedback and may continue to respond indefinitely (usually decaying) [27].

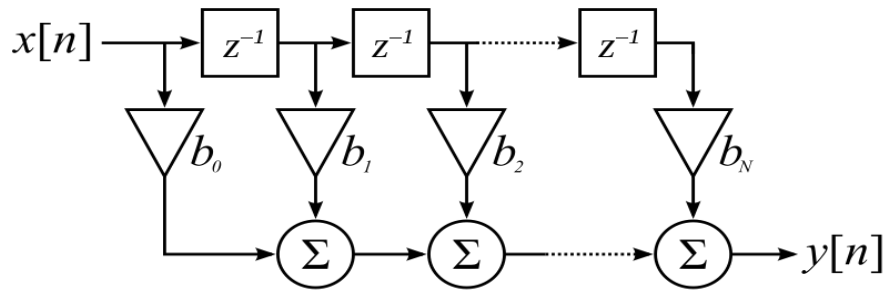


Fig. 3.8: General FIR filter.

The transfer function of FIR filter is defined as

$$H(z) = \sum_{n=0}^N b_n z^{-n}$$

Where b_n are the filter coefficients, also known as tap weights, which make up the impulse response.

Properties of FIR Filter:

An FIR filter has a number of useful properties:

- Require no feedback. This means that any rounding errors are not compounded by summed iterations. The same relative error occurs in each calculation. This also makes implementation simpler.
- Are inherently stable. This is due to the fact that, because there is no required feedback, all the poles are located at the origin and thus are located within the unit circle (the required condition for stability in a discrete, linear-time invariant system).
- They can easily be designed to be linear phase by making the coefficient sequence symmetric; linear phase, or phase change proportional to frequency, corresponds to equal delay at all frequencies. This property is sometimes desired for phase-sensitive applications, for example data communications, crossover filters, and mastering [27].

The main disadvantage of FIR filters is that considerably more computation power in a general purpose processor is required compared to an IIR filter with similar sharpness or selectivity, especially when low frequency (relative to the sample rate) cutoffs are needed.

However many digital signal processors provide specialized hardware features to make FIR filters approximately as efficient as IIR for many applications.

3.5 UNDERSTANDING POLE ZERO PLOT AND STABILITY ANALYSIS

A system is characterized by its poles and zeros in the sense that they allow reconstruction of the input/output differential equation. In general, the poles and zeros of a transfer function may be complex, and the system dynamics may be represented graphically by plotting their locations on the complex s -plane, whose axes represent the real and imaginary parts of the complex variable s . Such plots are known as pole-zero plots. It is usual to mark a zero location by a circle (\circ) and a pole location a cross (\times). The location of the poles and zeros provide qualitative insights into the response characteristics of a system [28].

A pole-zero plot can represent either a continuous-time (CT) or a discrete-time (DT) system. For a CT system, the plane in which the poles and zeros appear is the s plane of the Laplace transform. In this context, the parameter s represents the complex angular frequency, which is the domain of the CT transfer function. For a DT system, the plane is the z plane, where z represents the domain of the Z-transform.

In order for a linear system to be stable, all of its poles must have a negative real part, that is they must all lie within the left-half of the s -plane. An “unstable” pole, lying in the right half of the s -plane, generates a component in the system homogeneous response that increases without bound from any finite initial conditions. A system having one or more poles lying on the imaginary axis of the s -plane has non-decaying oscillatory components in its homogeneous response, and is defined to be marginally stable.

The region of convergence (ROC) for a given DT transfer function is a disk or annulus which contains no poles. In general, the ROC is not unique, and the particular ROC in any given case depends on whether the system is causal or anti-causal [28].

- If the ROC includes the unit circle, then the system is Bounded Input, Bounded Output (BIBO) stable.

- If the ROC extends outward from the pole with the largest (but not infinite) magnitude, then the system is causal.
- If the ROC extends inward from the pole with the smallest (nonzero) magnitude, then the system is anti-causal.

The ROC is usually chosen to include the unit circle since it is important for most practical systems to have BIBO stability [28].

CHAPTER 4

IIR DIGITAL FILTERS

Digital Filters are designed by using the values of both the past outputs and the present input, an operation brought about by convolution. If such a filter is subjected to an impulse then its output need not necessarily become zero. The impulse response of such a filter can be infinite in duration. Such a filter is called an Infinite Impulse Response filter or IIR filter. The infinite impulse response of such a filter implies the ability of the filter to have an infinite impulse response. This indicates that the system is prone to feedback and instability [29].

The report studies several different types of IIR filters including the Butterworth Filter, Chebyshev I & II Filters and Elliptic Low, High and Bandpass filters. IIR filters are designed essentially by the Impulse Invariance or the Bilinear Transformation method.

4.1 IMPULSE INVARIANCE

This procedure involves choosing the response of the digital filter as an equi-spaced sampled version of the analog filter [29].

1. Decide upon the desired frequency response
2. Design an appropriate analogue filter
3. Calculate the impulse response of this analogue filter
4. Sample the analogue filter's impulse response
5. Use the result as the filter coefficients

The impulse invariance method maps the left hand portion of the s-plane into the interior of the unit circle and the right hand portion of the s-plane to the exterior of the unit circle; hence each horizontal strip in the s-plane is overlaid onto the z-plane to form the digital system function from the analog system function. This is shown in Figure 4.1. Since any practical analog filter can never be bandlimited interference is a major consideration.

Due to the aliasing that arises in the sampling process the distortion in the frequency response is one of the major limiting factors of this implementation while its advantage lies in the fact that there is a linear relationship between the analog and digital frequency

response. Hence in order to prevent severe distortion due to the band limiting this method is restricted to the design of Low and Bandpass Filters [29].

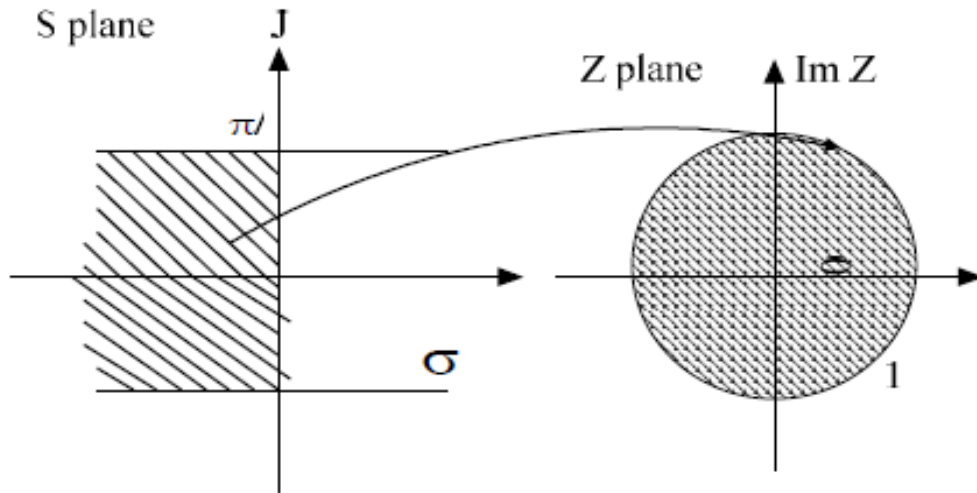


Fig. 4.1: Comparison of s and z plane using impulse invariance method [29].

4.2 BILINEAR TRANSFORMATION

There is another approximation technique of IIR digital filters from analog filters. It is called bilinear transformation method. This method is one of the best currently available methods for designing IIR digital filters due to simplicity and similarity of the frequency response of IIR digital filters to that of reference analog filters. The Bilinear Transformation method overcomes the effect of aliasing that is caused due to the analog frequency response containing components at or beyond the Nyquist Frequency [29].

The bilinear transform is a method of compressing the infinite, straight analogue frequency axis to a finite one long enough to wrap around the unit circle once only. This is also sometimes called frequency warping. This introduces a distortion in the frequency. This is undone by pre-warping the critical frequencies of the analog filter (cutoff frequency, centre frequency) such that when the analog filter is transformed into the digital filter, the designed digital filter will meet the desired specifications [29].

Bilinear Transformation:

Consider an analog filter:

$$H(s) = \frac{b}{s-a} \quad (4.1)$$

This system can be characterized by a differential equation

$$\frac{dy(t)}{dt} + ay(t) = bx(t) \quad (4.2)$$

Suppose we approximate the integral rather than the derivative

$$y(t) = \int_{t_0}^t y' \tau d\tau + y(t_0) \quad (4.3)$$

We can approximate the integral by using the Trapezoidal formula

$$y(nT) = \frac{T}{2} [y'(nT) + y'(nT - T)] + y(nT - T) \quad (4.4)$$

From the differential equation we can substitute for $y'(t)$

$$y'(nT) = -ay(nT) + bx(nT) \quad (4.5)$$

We can substitute this in the trapezoidal rule and write

$$\left[1 + \frac{aT}{2}\right] y(n) - \left[1 - \frac{aT}{2}\right] y(n-1) = \frac{bT}{2} [x(n) + x(n-1)] \quad (4.6)$$

The Z-transform of this gives:

$$\left[1 + \frac{aT}{2}\right] Y(z) - \left[1 - \frac{aT}{2}\right] z^{-1} Y(z) = \frac{bT}{2} [(1 + z^{-1})X(z)] \quad (4.7)$$

Which is simplified to

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b}{\frac{2(1-z^{-1})}{T(1+z^{-1})} + a} \quad (4.8)$$

Clearly the mapping is as follows

$$H(z) = H(s) \Big|_{s = \frac{2(1-z^{-1})}{T(1+z^{-1})}} \quad (4.9)$$

This mapping is known as bilinear transformation.

By solving this equation for y we obtain

$$s \Leftrightarrow \frac{2(1-z^{-1})}{T(1+z^{-1})} \quad (4.10)$$

This transformation is known as the Bilinear or Tustin Transformation. The Laplace transforms in the filter expressions are replaced by the corresponding z -transforms. Replacing $s = \sigma + j\Omega$ and performing algebraic manipulations, substituting $z = e^{j\omega}$ we get

$$\omega = 2 \tan^{-1}(\Omega T / 2) \quad (4.11)$$

It can be seen that analog dc ($s = 0$) maps to digital dc ($z = 1$) and the highest analog frequency ($s = \infty$) maps to the highest digital frequency ($z = -1$). It is easy to show that the entire $j\omega$ axis in the s plane is mapped exactly once around the unit circle in the z plane.. Therefore, it does not alias. With $(2/T)$ real and positive, the left-half s plane maps to the interior of the unit circle, and the right-half s plane maps outside the unit circle.

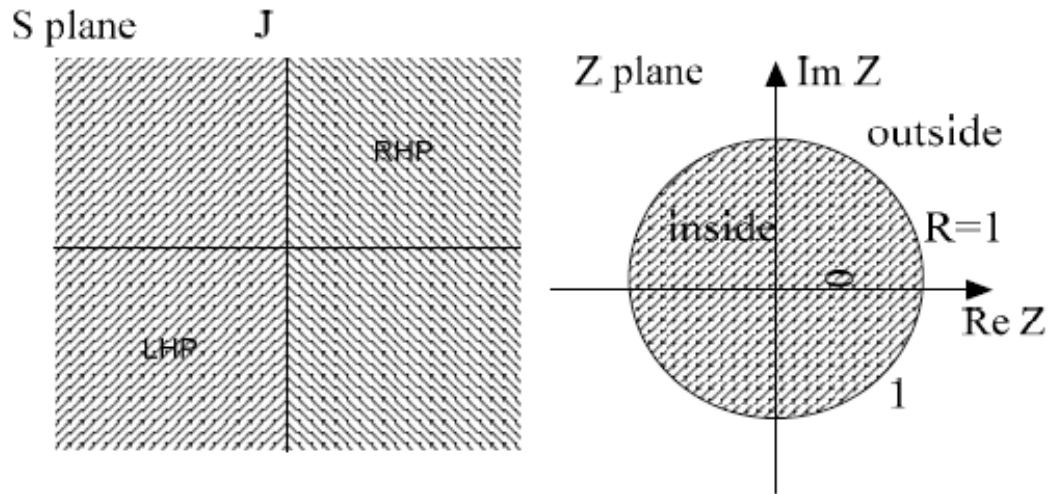


Fig 4.2: Comparison of s and z plane using bilinear transformation technique [29].

The constant provides one remaining degree of freedom that can be used to map any particular finite frequency the $j\omega$ axis in the s plane to a particular desired location on the unit circle $e^{j\omega}$ in the z plane. All other frequencies will be warped. In particular, approaching half the sampling rate, the frequency axis compresses more and more [29].

Filters having a single transition frequency, such as lowpass or highpass filters, map beautifully under the bilinear transform; you simply map the cut-off frequency where it belongs, and the response looks great. In particular, "equal ripple" is preserved for optimal filters of the elliptic and Chebyshev types because the values taken on by the frequency response are identical in both cases; only the frequency axis is warped [29].

4.3 TRANSFER FUNCTION OF AN IIR FILTER

Equation 4.12 defines the direct-form transfer function of an IIR filter.

$$H(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_{N_b-1} z^{-(N_b-1)}}{1 + a_1 z^{-1} + \dots + a_{N_a-1} z^{-(N_a-1)}} \quad (4.12)$$

A filter implemented by directly using the structure defined by Equation 4.12. Where a_n and b_n are the reverse and forward coefficients of the IIR filter.

It can be written in the form of general difference equation as follows

$$y_i = \frac{1}{a_0} \left(\sum_{j=0}^{N_b-1} b_j x_{i-j} - \sum_{k=0}^{N_a-1} a_k y_{i-k} \right) \quad (4.13)$$

where b_j is the set of forward coefficients, N_b is the number of forward coefficients, a_k is the set of reverse coefficients, and N_a is the number of reverse coefficients. Equation (4.13) describes a filter with an impulse response of theoretically infinite length for nonzero coefficients. However, in practical filter applications, the impulse response of a stable IIR filter decays to near zero in a finite number of samples.

In most IIR filter designs and all of the MATLAB IIR filters, coefficient a_0 is 1. The output sample at the current sample index i is the sum of scaled current and past inputs and scaled past outputs, as shown by Equation 4.14

$$y_i = \left(\sum_{j=0}^{N_b-1} b_j x_{i-j} - \sum_{k=0}^{N_a-1} a_k y_{i-k} \right) \quad (4.14)$$

where x_i is the current input, x_{i-j} is the past inputs, and y_{i-k} is the past outputs. IIR filters might have ripple in the passband, the stopband, or both.

4.4 IIR FILTER TYPES

Digital IIR filter designs come from the classical analog designs and include the following filter types [30]:

- Butterworth filters
- Chebyshev filters
- Chebyshev II filters, also known as inverse Chebyshev and Type II Chebyshev filters
- Elliptic filters, also known as Cauer filters

The IIR filter designs differ in the sharpness of the transition between the passband and the stopband and where they exhibit their various characteristics—in the passband or the stopband.

4.4.1 Butterworth Filters

Butterworth filters have the following characteristics:

- Smooth response at all frequencies

- Monotonic decrease from the specified cut-off frequencies
- Maximal flatness, with the ideal response of unity in the passband and zero in the stopband
- Half-power frequency, or 3 dB down frequency, that corresponds to the specified cut-off frequencies.

The transfer function for Butterworth filter is given by

$$B(\omega) = \frac{1}{[1+(\omega/\omega_0)^{2n}]^{\frac{1}{2}}} \quad (4.15)$$

Where n is the order of filter.

Butterworth

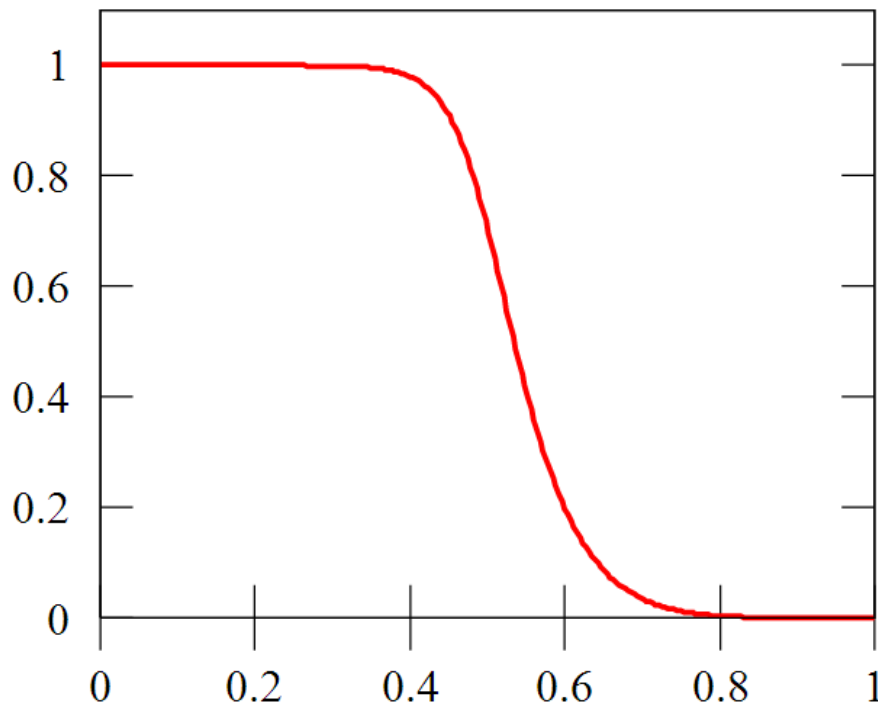


Fig. 4.3: Response of Butterworth Filter [30].

As shown in Figure 4.3, after specifying the cut-off frequency of a Butterworth filter, MATLAB sets the steepness of the transition proportional to the filter order. Higher order Butterworth filters approach the ideal lowpass filter response. Butterworth filters do not

always provide a good approximation of the ideal filter response because of the slow rolloff between the passband and the stopband.

4.4.2 Chebyshev Filters

Chebyshev filters have the following characteristics [30]:

- Minimization of peak error in the passband
- Equiripple magnitude response in the passband
- Monotonically decreasing magnitude response in the stopband
- Sharper rolloff than Butterworth filters

Compared to a Butterworth filter, a Chebyshev filter can achieve a sharper transition between the passband and the stopband with a lower order filter. The sharp transition between the passband and the stopband of a Chebyshev filter produces smaller absolute errors and faster execution speeds than a Butterworth filter.

Chebyshev type 1

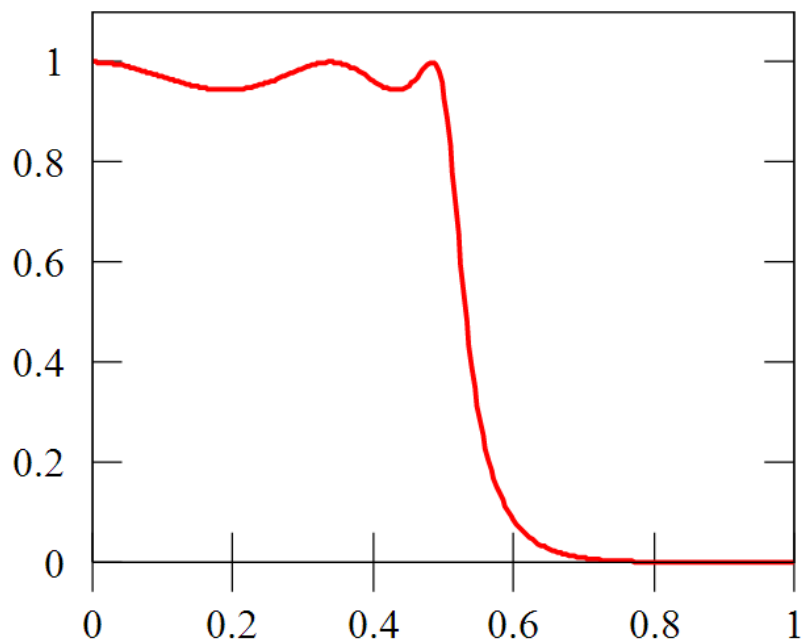


Fig. 4.4: Response of Chebyshev filter [30].

Figure 4.4 shows the frequency response of a lowpass Chebyshev filter. In Figure 4.4, the maximum tolerable error constrains the equiripple response in the passband. Also, the sharp rolloff appears in the stopband. The frequency response of the filter is given by

$$|H(\Omega)|^2 = \left(1 + \varepsilon^2 T_n^2(\Omega/\Omega_p)\right)^{-1} \quad (4.16)$$

Where ε is a parameter of the filter related to ripple present in the passband and $T_n(x)$ is the Nth- order Chebyshev polynomial defined as

$$T_n = \cos(N \cos^{-1} x) \quad \text{when } |x| \leq 1$$

$$T_n = \cos(N \cosh^{-1} x) \quad \text{when } |x| \geq 1$$

4.4.3 Chebyshev II filters or Inverse chebyshev filters

Chebyshev II filters have the following characteristics [30]:

- Minimization of peak error in the stopband
- Equiripple magnitude response in the stopband
- Monotonically decreasing magnitude response in the passband
- Sharper rolloff than Butterworth filters

Chebyshev II filters are similar to Chebyshev filters. However, Chebyshev II filters differ from Chebyshev filters in the following ways:

- Chebyshev II filters minimize peak error in the stopband instead of the passband. Minimizing peak error in the stopband instead of the passband is an advantage of Chebyshev II filters over Chebyshev filters. Chebyshev II filters have an equiripple magnitude response in the stopband instead of the passband.
- Chebyshev II filters have a monotonically decreasing magnitude response in the passband instead of the stopband

In Figure 4.5, the maximum tolerable error constrains the equiripple response in the stopband. Also, the smooth monotonic rolloff appears in the stopband. Chebyshev II filters have the same advantage over Butterworth filters that Chebyshev filters have a sharper transition between the passband and the stopband with a lower order filter, resulting in a smaller absolute error and faster execution speed.

Chebyshev type 2

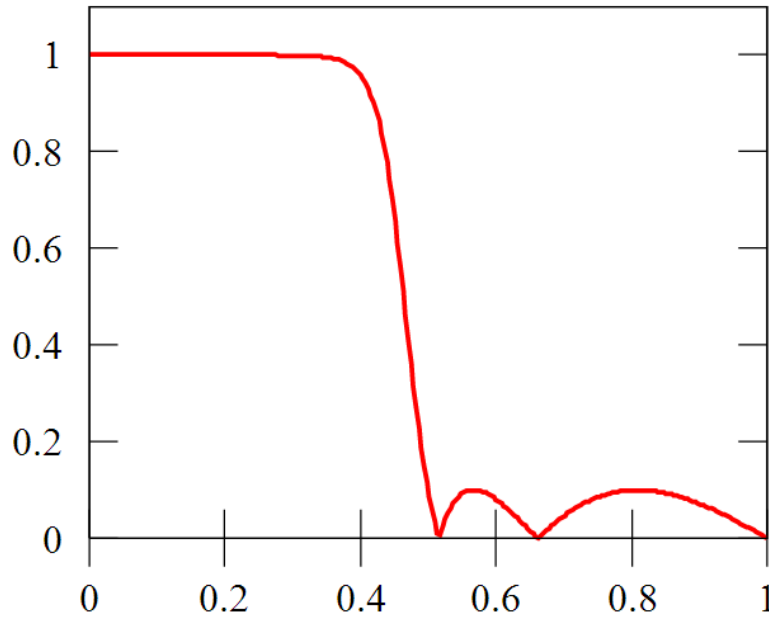


Fig. 4.5: Response of Inverse Chebyshev filter [30].

4.4.4 Elliptic Filters

Elliptic filters have the following characteristics [30]:

- Minimization of peak error in the passband and the stopband
- Equiripples in the passband and the stopband

Compared with the same order Butterworth or Chebyshev filters, the elliptic filters provide the sharpest transition between the passband and the stopband, which accounts for their widespread use.

In Figure 4.6, the same maximum tolerable error constrains the ripple in both the passband and the stopband. Also, even low-order elliptic filters have a sharp transition edge. The transfer function is given by

$$|H(\Omega)|^2 = \left(1 + \varepsilon^2 U_n(\Omega/\Omega_c)\right)^{-1} \quad (4.17)$$

where $U_n(x)$ is the Jacobian elliptic function of order N and ε is a constant related to passband ripple. They provide a realization with the lowest order for a particular set of conditions.

Elliptic

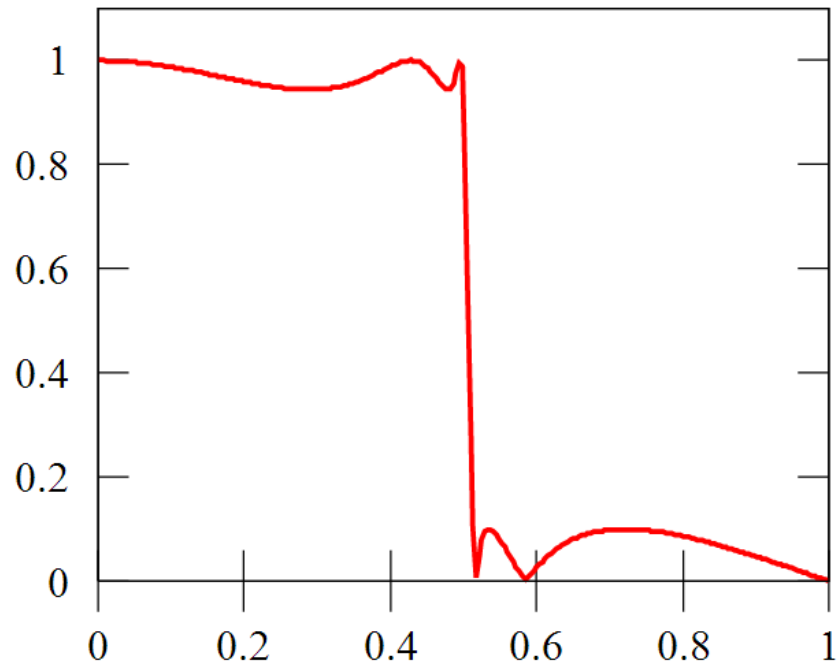


Fig. 4.6: Response of Elliptic Filter [30].

4.5 SELECTING A DIGITAL FILTER DESIGN

Digital filters are selected according to the following application [31]:

- Does the analysis require a linear-phase response?
- Can the analysis tolerate ripples?
- Does the analysis require a narrow transition band?

Figure 4.7 works as a guideline for selecting the appropriate filter for an analysis application.

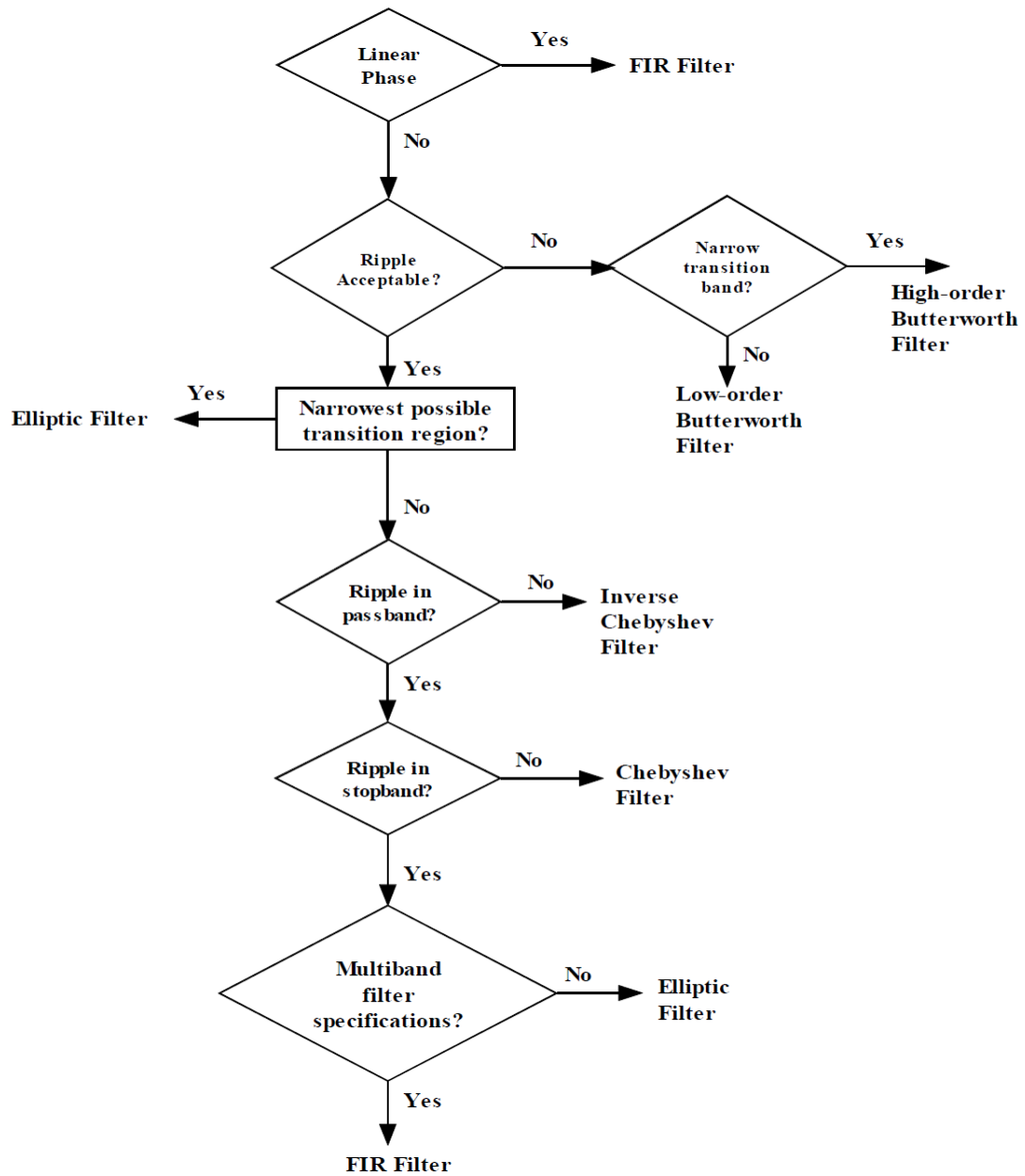


Fig. 4.7: Flowchart for selecting Filter [31].

CHAPTER 5

MATHEMATICAL MODELLING OF IIR DIGITAL FILTERS

In many applications, differentiation is followed by low-pass filtering. Differentiation is used to abstract data about swift transients in the signal. Low-pass filters are used to reject noise frequencies greater than the cut-off frequencies of the signal. Low-pass filtering and differentiation are realized as a single low-pass differentiator filter or by cascading low-pass filter and a differentiating filter. The resulting low-pass differentiators are called IIR differentiators.

Analog differentiators are often obtained by inverting the transfer functions of analog integrators, or vice versa. Extending this concept to non-minimum phase-digital integrators yields unstable differentiators. The stabilization approach developed below will be employed. The approach, as applied in this work, consists of the following four steps [11]:

- Obtain the discrete-time transfer function, using the z -transform, of a digital integrator that has the desired range and accuracy. This would approximate the analog transfer function $1/s$, using the Laplace transform, of an integrator.
- Invert the discrete-time transfer function of the integrator obtained in (5.3) to obtain the digital approximation of the analog transfer function s . This approximation constitutes a proposed s -to- z transform [11].
- Stabilize the resulting discrete-time transfer function obtained in (5.7), by reflecting its poles that lie outside the unit circle at radius r to inside the unit circle at a radius of $1/r$. This corresponds to adding a zero at r and a pole at $1/r$, which is equivalent to multiplying the original transfer function by an all-pass filter. For the case of a pole of order n at infinity, introduce a pole of order n at zero. In this latter case, when a zero is added at infinity, a pole is also obtained at zero [11].
- Compensate the resulting change in the magnitude of the discrete-time transfer function, resulting from step (3), by multiplying the resulting transfer function by $1/r$ for each reflected pole. No magnitude compensation is needed for the case of a pole of order n at infinity since the added pole of order n at the origin of the z -plane will have a magnitude of one when it is evaluated on the unit circle of the z -plane.

A minimum phase filter's zeros may lie anywhere inside the unit circle of the z-plane. Zeros are permitted to lie on the unit circle provided they are simple (i.e., they are of order one).

In our design approach we are considering linear order low power IIR digital differentiator design and fractional order IIR differentiator design through inverse integration rule and later on comparing their magnitude and phase response.

5.1 INTEGER ORDER IIR DIGITAL DIFFERENTIATOR:

The basic concept came from observing that the ideal integrator response lies between the responses of the traditional trapezoidal and Simpson integrators. Thus it seems reasonable that interpolating the above two rules could yield integrators that better approximate the ideal integrator [6].

5.1.1 BASIC INTEGRATION RULES:

(1) Simpson Integrator

(2) Trapezoidal Integrator

5.1.1.1 Simpson Integrator: In numerical analysis, Simpson's rule is a method for numerical integration, the numerical approximation of definite integrals.

Specifically, it is the following approximation:

$$\int_a^b f(x)dx = \frac{(b-a)}{6} \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right] \quad (5.1)$$

If the interval of integration [a, b] is in some sense "small", then Simpson's rule will provide an adequate approximation to the exact integral. By small, it means that the function being integrated is relatively smooth over the interval [a, b]. For such a function, a smooth quadratic interpolant like the one used in Simpson's rule will give good results.

However, it is often the case that the function we are trying to integrate is not smooth over the interval. Typically, this means that either the function is highly oscillatory, or it lacks derivatives at certain points. In these cases, Simpson's rule may give very poor results. One common way of handling this problem is by breaking up the interval into a number of small subintervals. Simpson's rule is then applied to each subinterval, with the results being summed to produce an approximation for the integral over the entire interval. This sort of approach is termed the *composite Simpson's rule*.

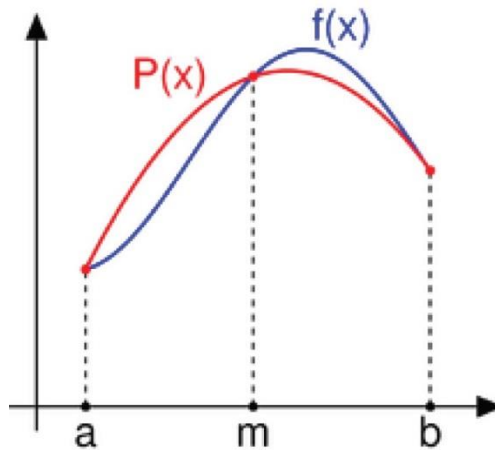


Fig. 5.1: Simpson's rule can be derived by approximating the integrand $f(x)$ by the quadratic interpolant $P(x)$ [32].

5.1.1.2 Trapezoidal Integrator: In numerical analysis, the trapezoidal rule (also known as the trapezoid rule or trapezium rule) is a technique for approximating the definite integral

$$\int_a^b f(x)dx$$

The trapezoidal rule works by approximating the region under the graph of the function $f(x)$ as a trapezoid and calculating its area. It follows that

$$\int_a^b f(x) \approx (b - a) \frac{f(a) + f(b)}{2} \quad (5.2)$$

Simpson's rule is another member of the same family, and in general has faster convergence than the trapezoidal rule for functions which are twice continuously differentiable, though not in all specific cases. However for various classes of rougher functions (ones with weaker smoothness conditions), the trapezoidal rule has faster convergence in general than Simpson's rule.

Moreover, the trapezoidal rule tends to become extremely accurate when periodic functions are integrated over their periods, which can be analysed in various ways. For various classes of functions that are not twice-differentiable, the trapezoidal rule has sharper bounds than Simpson's rule.

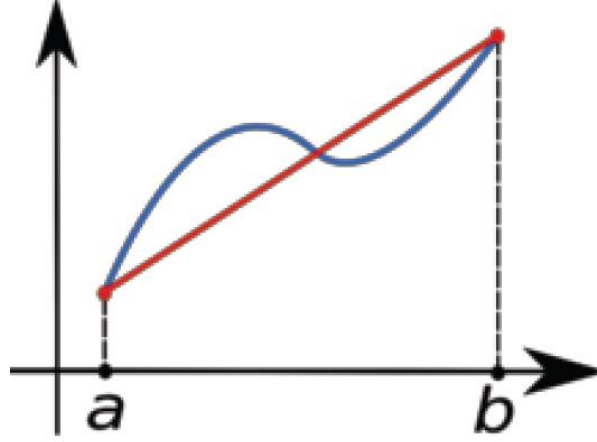


Fig. 5.2 The function $f(x)$ (in blue) is approximated by a linear function (in red) [32].

5.1.2 INTEGRATOR DESIGN: It was shown in [7] that the magnitude of the frequency response of the ideal integrator lies between the Simpson and the trapezoidal integrators. Interpolating the two integrators in a manner similar to that in [11] and to that used to obtain the Simpson rule from the midpoint and trapezoidal rule yields

$$H(z) = a H_{Rect}(z) + (1 - a) H_{Trap}(z) \quad (5.3)$$

where a is a real valued parameter defined on the closed interval $(0, 1)$. $H_{Rect}(z)$ and $H_{Trap}(z)$ are the transfer functions of the Simpsons and the trapezoidal integrators, respectively, and are shown below:

$$H_{rect}(z) = \frac{T(z^2+4z+1)}{3(z^2-1)} \quad (5.4)$$

$$H_T(z) = \frac{T(z+1)}{2(z-1)} \quad (5.5)$$

Substituting (5.4) and (5.5) in (5.3) and simplifying yields the following transfer function for the novel class of non-minimum phase integrators:

$$H(z) = \frac{T(3-a)\{z^2 + \frac{z(3+a)}{3-a}z + 1\}}{6(z^2-1)} \quad (5.6)$$

The numerator of (5.6) may be factored to yield

$$H(z) = \frac{T(3-a)[(z+r_1)(z+r_2)]}{6(z^2-1)} \quad (5.7)$$

$$\text{Where } r_1 = \frac{(3+a+2\sqrt{3a})}{(3-a)} \text{ and } r_2 = \frac{(3+a-2\sqrt{3a})}{(3-a)} \quad (5.8)$$

This class of integrators has the property that its zeros are reciprocal pairs around the unit circle in the z -plane, since $r_1 = 1/r_2$ [11].

5.1.3 IIR-TYPE INTEGER ORDER DIGITAL DIFFERENTIATOR DESIGN:

Firstly, we can obtain a family of new integer-order digital differentiators from the digital integrators introduced in Section 5.1.1. Direct inversion of $H(z)$ will give an unstable filter since $H(z)$ has a zero r_1 . By reflecting the r_1 to $1/r_1$, i.e. r_2 , we have the stabilized response $\mathcal{H}(z)$ [14]

$$\mathcal{H}(z) = \frac{KT(3-a)(z+r_2)^2}{6(z^2-1)} \quad (5.9)$$

To determine K , let the final value of the impulse responses of $H(z)$ and $\mathcal{H}(z)$ be the same, i.e., $\lim_{z \rightarrow 1} (z-1)H(z) = \lim_{z \rightarrow 1} (z-1)\mathcal{H}(z)$, which gives $K=r_1$. Therefore, a new family of the digital differentiators can be given by

$$w(z) = \frac{1}{\mathcal{H}(z)} = \frac{6(z^2-1)}{r_1 T(3-a)(z+r_2)^2} = \frac{r_2 6(z^2-1)}{T(3-a)(z+r_2)^2} \quad (5.10)$$

The approach consists of inverting the transfer function of an integrator, stabilizing the resulting transfer function by reflecting the poles that lie outside the unit circle in the z -plane to inside the unit circle, and compensating the magnitude appropriately. The compensation notes that if a pole that lies at a radius r is replaced by a pole that lies at a radius of $1/r$, the magnitude of the resulting transfer function will be multiplied by r [14].

Thus to compensate for the change in magnitude, the resulting transfer function should be multiplied by $1/r$. The magnitude of the frequency response of the resulting differentiator has the same range and accuracy as the integrator.

5.2 IIR-TYPE FRACTIONAL ORDER DIGITAL DIFFERENTIATOR DESIGN:

Fractional order systems are systems that are described by fractional differential equations in which the integer order n of the derivative operator $D^n = d^n/dt^n$ is generalized to real or complex order α , such that one can define the operator $D^\alpha = d^\alpha/dt^\alpha$. The fractional

order differentiators or integrators are defined, in the Laplace domain, by the following transfer function

$$H(s) = s^\alpha \quad (5.11)$$

where s is the Laplace operator. These systems are used to calculate the fractional order time derivative and integral of an input signal. Many algorithms have been developed to best approximate the fractional order operator s^α with analogue or digital integer models. In general, there are two discretization methods: direct and indirect methods [33].

In indirect methods, a rational continuous-time models approximation of s^α is first developed, then the resulting s -function is discretized using an appropriate s -to- z transform. Whereas the direct methods substitute for s^α an appropriate s -to- z transform, then the resulting non- rational z -function is approximated with rational one. In this work a family of second order mapping functions obtained by inverting and stabilizing the combination of the numerical integrator rules of trapezoidal and Simpson, have been used with CFE to design rational approximation of s^α [33].

Also in this work a new second order s -to- z transform, (new mapping function (NMF)) obtained by inversion and stabilizing the new Simpson integrator designed using conventional Simpson-integrator rule and fractional delay filter is proposed.

By using equation obtained from [18] Al-Alaoui operator for discretization technique is given as

$$s \approx \frac{8}{7T} \frac{(z-1)}{(z+\frac{1}{7})} \quad (5.12)$$

Now using this operator as to find the fractional order response of the system we put the value of ' s ' in equation (5.11), we obtain a new transfer function for digital fractional order differentiator which is

$$s^\alpha = H^\alpha(z) = \left(\frac{8}{7T} \frac{(z-1)}{(z+\frac{1}{7})} \right)^\alpha \quad (5.13)$$

where value of α lies between 0 to 1.

From [19] the NMF function the new digital fractional order differentiator is given as

$$s^\alpha \approx H^\alpha(z) = \left(\frac{18}{7T(-p_1)}\right)^\alpha \left((1 - z_1 z^{-1})^\alpha (1 - z_2 z^{-1})^\alpha / \left(1 + \frac{z^{-1}}{p_1}\right)^\alpha (1 - p_2 z^{-1})^\alpha \right) \quad (5.14)$$

Clearly $H^\alpha(z)$ in (5.14) is irrational function; the aim of this work is to transform it into rational function using PSE modelling technique by modelling the impulse response of the DFOD given in (5.14).

CHAPTER 6

SIMULATION RESULTS

6.1 INTEGER ORDER IIR DIGITAL DIFFERENTIATOR DESIGN:

By using MATLAB function on the inverse integral function derived in equation (5.10) $w(z) = \frac{1}{\mathcal{H}(z)} = \frac{6(z^2-1)}{r_1 T(3-a)(z+r_2)^2} = \frac{r_2 6(z^2-1)}{T(3-a)(z+r_2)^2}$ and by varying the values of 'a' from '0' to '1' with 0.2 increment the following plots are obtained. The new differentiator has higher accuracy or greater range of frequency than the traditional differentiators. The magnitude plot is obtained as follows shown in fig. 6.1.

The value of 'a' dictates that the magnitude response is between simpson and trapezoidal response. The resulting low-pass differentiator compare favourably with the 10 point low pass differentiators proposed by Oppenheim and Schafr which has an error of about 12% for a range of 0.8 of the Nyquist frequency. The plots reveal a reduction in the magnitude roll-off for the as the value of 'a' approaches to unity.

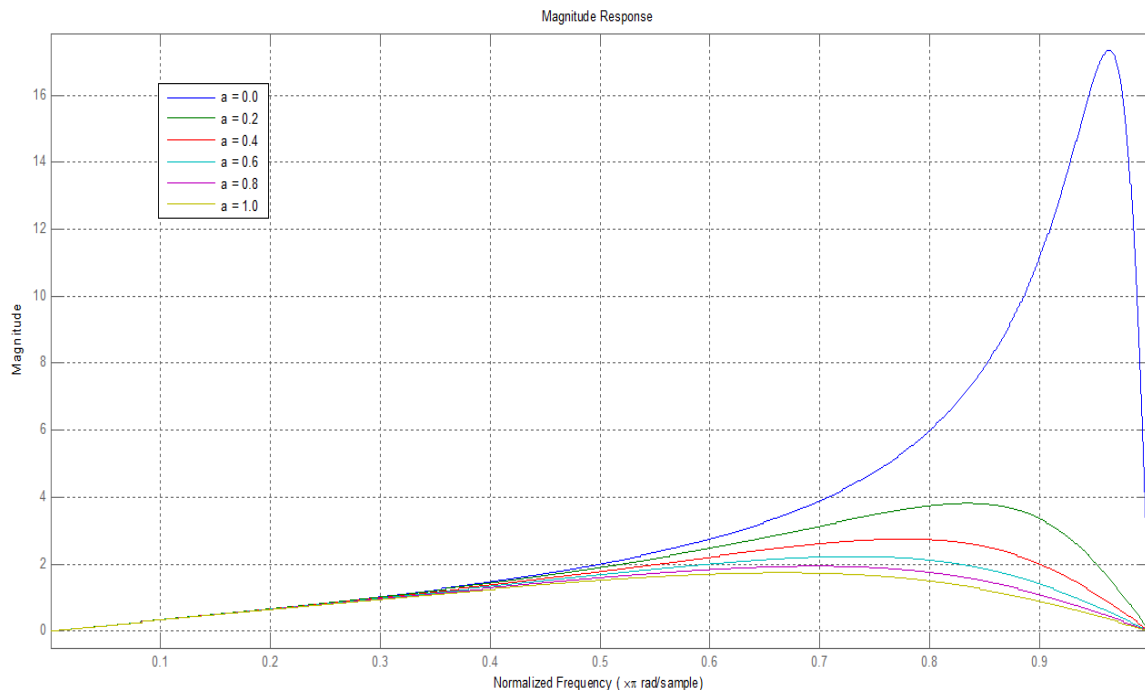


Fig. 6.1: Magnitude response of obtained differentiator lying between simpson and trapezoidal response.

Table 1: Magnitude response at normalized frequency at different values of ‘a’

Normalised frequency	a=0.0	a=0.2	a=0.4	a=0.6	a=0.8	a=1.0
0.0	0	0	0	0	0	0
0.1	0.377	0.321	0.317	0.316	0.315	0.315
0.2	0.651	0.653	0.644	0.637	0.633	0.627
0.3	1.018	1.013	0.986	0.971	0.955	0.940
0.4	1.452	1.419	1.357	1.318	1.274	1.236
0.5	1.994	1.890	1.767	1.671	1.580	1.502
0.6	2.736	2.463	2.199	2.000	1.830	1.698
0.7	3.883	3.127	2.595	2.218	1.936	1.710
0.8	5.968	3.734	2.720	2.122	1.744	1.471
0.9	11.159	3.348	1.990	1.396	1.084	0.882
0.95	16.716	2.057	1.108	0.749	0.589	0.458
1.0	0	0	0	0	0	0

Fig. 6.2 shows the phase responses for the low pass differentiators with a varying from 0 to 1 with an increment of 0.2. It is clear from the fig. 6.2 that the resulting phase responses are almost linear over the passbands as the value of ‘a’ increases. from the table when normalized frequency is 0.5 radians the value of the ideal linear phase is 83.37 , the phase response when a = 0.8 is 35.33 , whereas the phase when a=1.0 is 29.96, respectively, which corresponds to an improvement of 6%. The plots reveal improvement of the phase responses as the value of ‘a’ is increased.

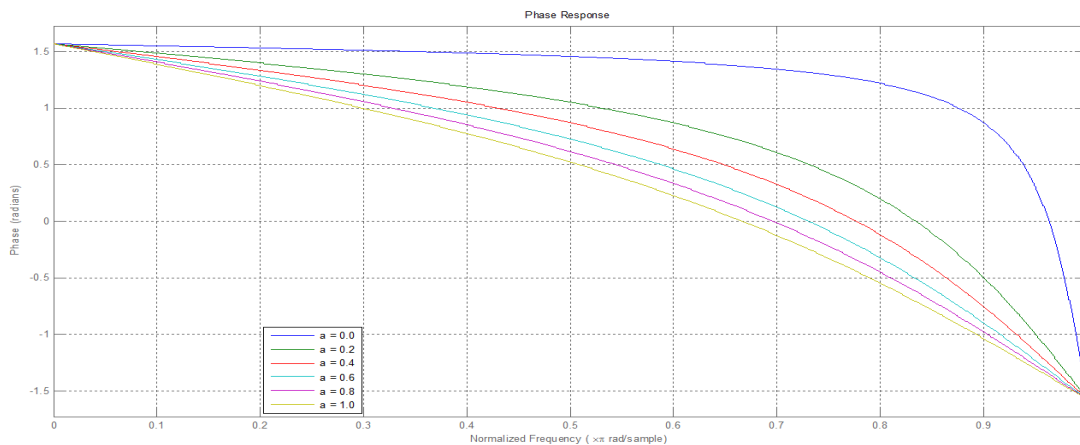


Fig. 6.2 Phase response of obtained differentiator lying between simpson and trapezoidal response.

Thus, the optimization procedure provides a trade-off between magnitude responses and phase responses.

Table 2: Phase response at normalized frequency at different values of ‘a’

Normalised Frequency	a=0.0	a=0.2	a=0.4	a=0.6	a=0.8	a=1.0
0.0	90	90	90	90	90	90
0.1	88.94	85.14	83.35	81.82	80.59	78.49
0.2	87.84	80.09	76.42	73.42	70.88	68.67
0.3	86.62	75.54	68.97	84.25	60.46	57.18
0.4	85.18	68.13	60.26	83.84	48.87	44.41
0.5	83.37	60.22	49.86	41.62	35.33	29.99
0.6	80.91	49.79	36.52	26.45	19.16	13.07
0.7	77.04	34.86	18.59	7.08	-0.87	-0.729
0.8	69.76	11.30	-6.69	-18.52	-25.68	-31.34
0.9	49.93	-28.49	-43.24	-51.49	-56.13	-59.56
1.0	-90	-90	-90	-90	-90	-90

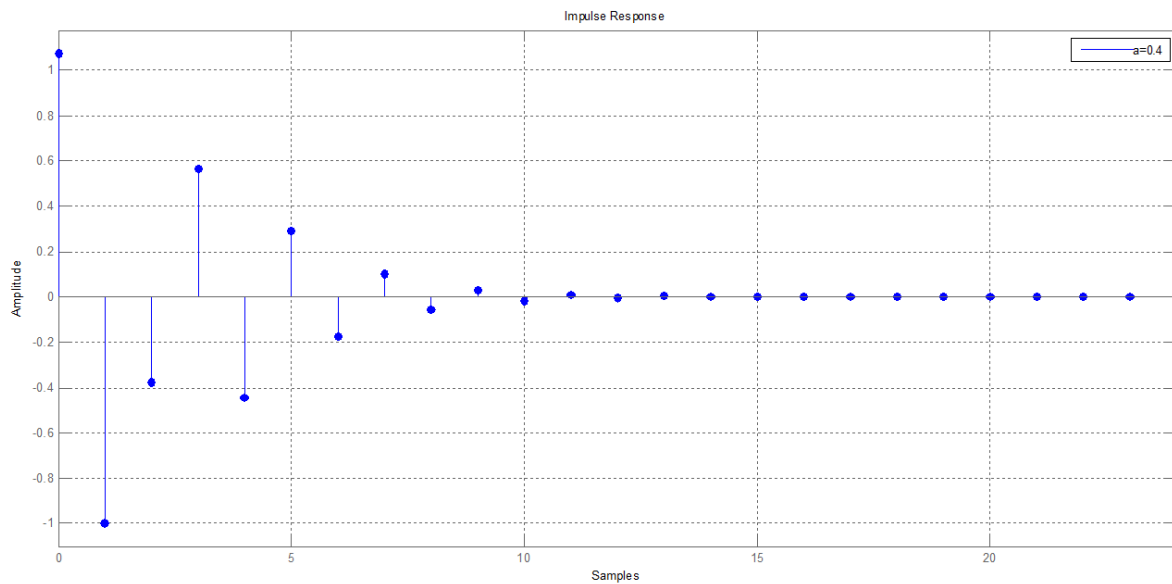


Fig. 6.3: Impulse response of obtained differentiator at a= 0.4.

The impulse response of the obtained differentiator at $a = 0.4$ is also performed and is shown in fig. 6.3. The impulse response presented in Fig. 6.3 for $a = 0.4$ has been modeled using the signal modeling technique of Steiglitz - McBride method implemented in Matlab by the function stmcb.m.

Fig. 6.4 shows the pole zero plot of the differentiator obtained by inverting the response of integrator obtained by equation 5.7. The plot clearly shows that the resulting differentiator is unstable as one of the poles is lying outside the unit circle. So the resulting approach is stabilized. In this approach the transfer function of the integrator is first inverted, and then the resulting transfer function is stabilized by reflecting the poles that lie outside the unit circle in the z-plane to inside the unit circle. After that the magnitude is compensated appropriately. In compensation technique pole lying at a radius r is replaced by a pole that lies at a radius of $1/r$ and the magnitude of the resulting transfer function is multiplied by r . Thus to compensate for the change in magnitude, the resulting transfer function is multiplied by $1/r$. The magnitude of the frequency response of the resulting differentiator has the same range and accuracy as the integrator.

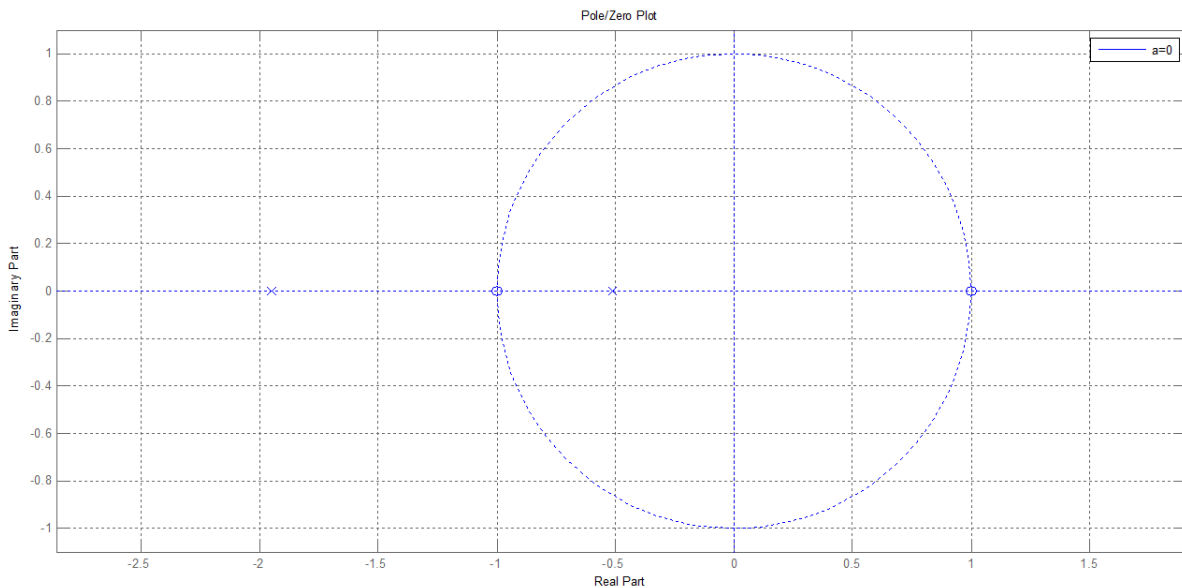


Fig. 6.4: Pole zero plot obtained by inverting response of integrator.

Fig. 6.5 shows the pole zero plot of stabilized differentiator. The diagram of pole-zeros of the obtained approximations (omitted for brevity) show that the poles and zeros are inside the

unit circle and distributed in alternating fashion on the real axis, then the approximations are stable and minimum phase.

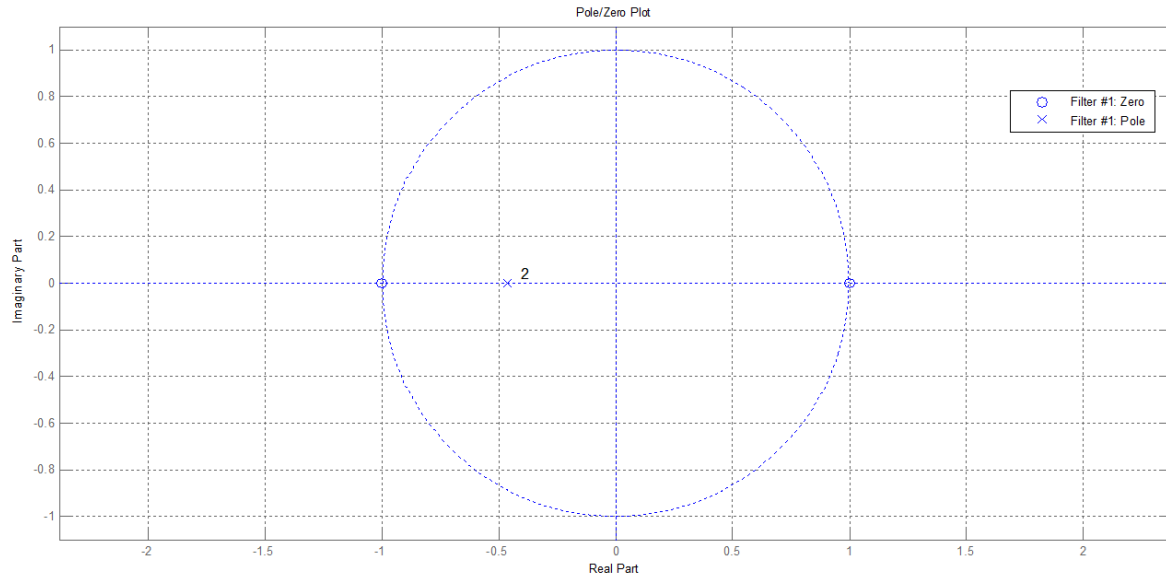


Fig. 6.5: pole zero plot of stabilized differentiator.

Above designed digital differentiator is now modeled to find the correct response by passing some test signal as shown in Fig. 6.6, where sin wave is applied at the input.

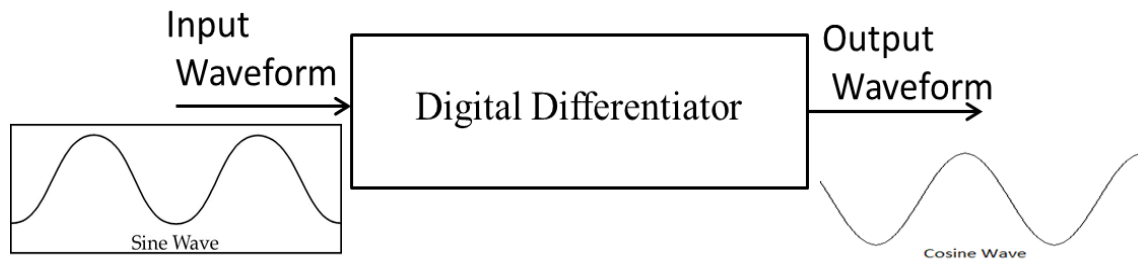


Fig. 6.6: Proposed model of digital differentiator.

Now as we can see from Fig. 6.6(a) a sin wave is applied to the input of proposed model. In Fig. 6.6(b) we add some random noise to the input signal. From the output signal obtained in Fig. 6.6(c) we can clearly see that the output signal phase is shifted by 90° . Also we can see from the figure that that the output signal is devoid of any noise which clearly shows that that the obtained model discussed in previous sections is giving the desired result.

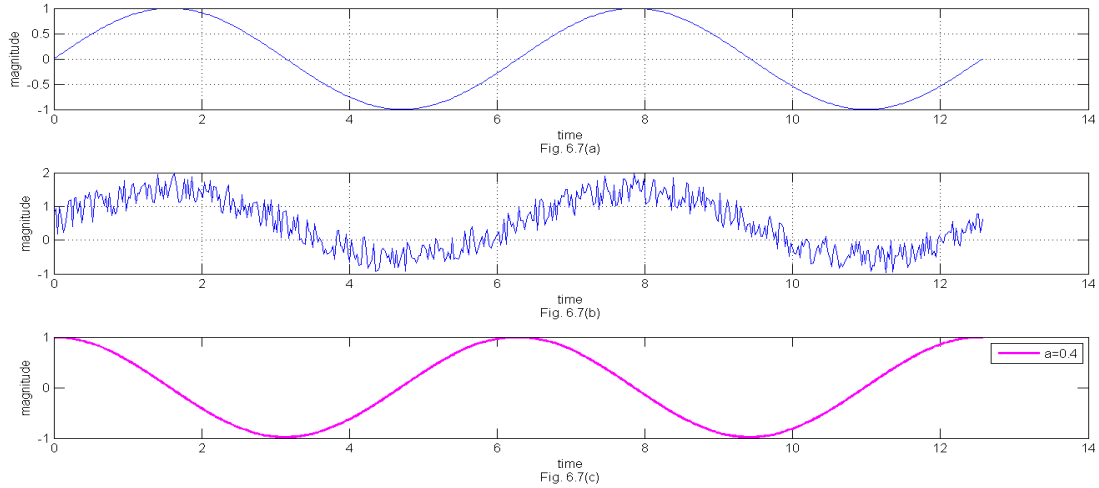


Fig. 6.7: Response of proposed model of digital differentiator

6.2 FRACTIONAL ORDER IIR DIGITAL DIFFERENTIATOR DESIGN:

Now for fractional order digital differentiator simulation results is presented for digital rational approximation (IIR filter) of continuous half differentiator $s^{0.5}$ sampled at $T=1s$, and the order of the obtained filter are chosen as 5. In Fig. 6.7 a comparison between the rational approximation of $s^{0.5}$ based on the NMF given in (5.14) and that based on the widely used, Al-Alaoui operator, using PSE modelling approach obtained in (5.13) is done. The magnitude response of the obtained approximations (IIR filters) for $s^{0.5}$ is depicted in Fig.6.7. As it can be seen from Fig. 6.7 the NMF improves the high frequency magnitude response comparatively to Al-Alaoui operator.

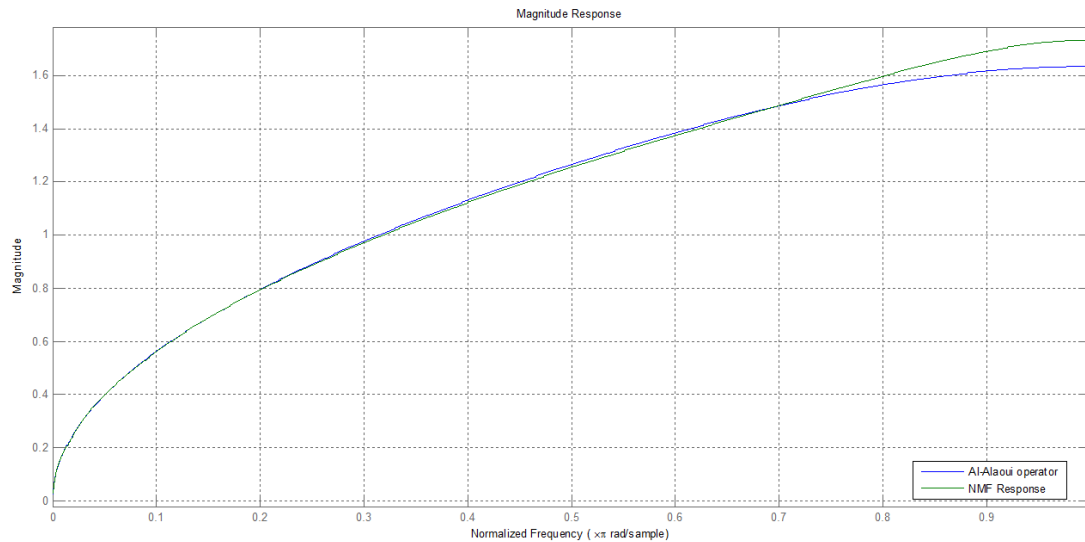


Fig. 6.8: Magnitude response of Al-Alaoui operator and NMF

Fig. 6.8 shows the comparative waveform for phase response for NMF response and the Al - Alaoui operator. From the figure we can see that NMF based approximation has better phase response than that of the Al-Alaoui operator based approximation.

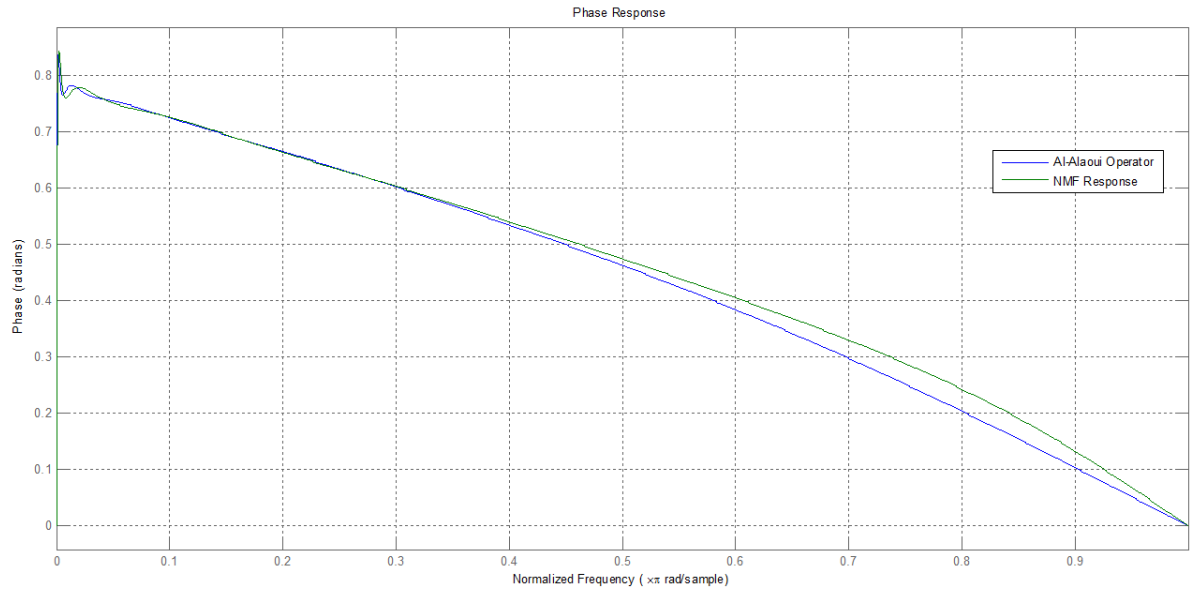


Fig. 6.9: Phase response of Al-Alaoui operator and NMF

The impulse response presented in Fig.2 for $\alpha = 0.5$ has been modeled using the signal modeling technique of Steiglitz - McBride method implemented in Matlab by the function stmcb.m.

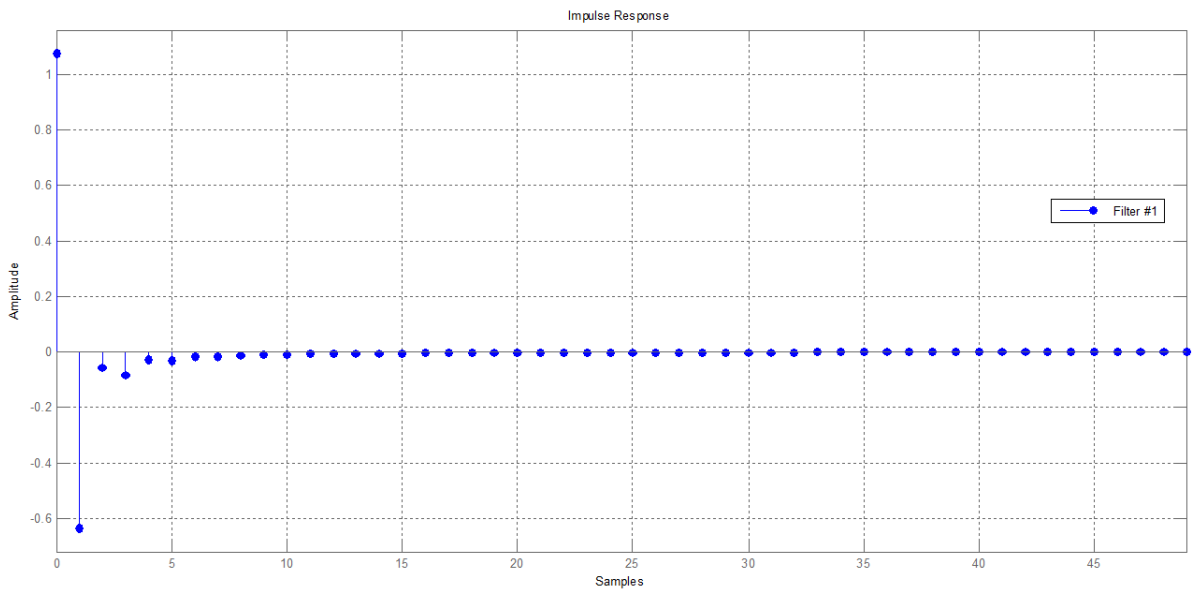


Fig. 6.10: Impulse response of fractional order operator obtained in equation 5.14

Fig. 6.10 shows the diagram of pole-zeros of the obtained approximations showing that the poles and zeros are inside the unit circle and are distributed in alternating fashion on the real axis so the approximations are stable and minimum phase.

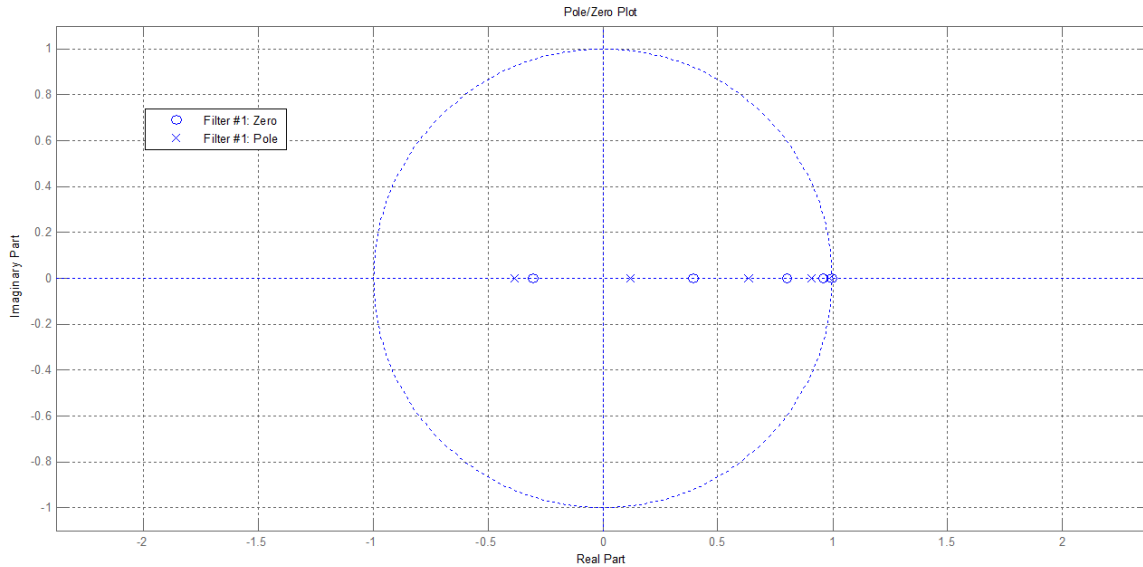


Fig 6.11: Pole zero plot of Al-Alaoui operator

Fig. 6.11 shows the minimum number of adders and multipliers required for the implementation of the integer order digital differentiator obtained after the simulation of proposed design of equation (5.9) in MATLAB.

```

Discrete-Time IIR Filter (real)
-----
Filter Structure      : Direct-Form II Transposed
Numerator Length     : 3
Denominator Length   : 3
Stable                : Yes
Linear Phase         : No

Implementation Cost
Number of Multipliers : 5
Number of Adders      : 3
Number of States      : 2
Multiplications per Input Sample : 5
Additions per Input Sample : 3
    
```

Fig. 6.12: Hardware requirement for integer order digital differentiator

Fig. 6.11 shows the minimum number of adders and multipliers required for the implementation of the fractional order digital differentiator obtained after the simulation of proposed design of equation (5.13) and (5.14) in MATLAB. From the consideration of the two designs we can clearly see that the amount of hardware required for the implementation of fractional order digital differentiator is more than that of integer order digital differentiator leading to complexity in design of fractional order. But we have to also keep in consideration that the magnitude response and phase response obtained at high frequency is more accurate for fractional order. So as per our design requirement we have to keep a trade-off between integer order and fractional order. For faster implementation and cost efficiency we can choose integer order and for better and accurate results we can choose fractional order.

```

% Filter #1
% -----
Discrete-Time IIR Filter (real)
-----
Filter Structure   : Direct-Form II Transposed
Numerator Length  : 6
Denominator Length : 6
Stable            : Yes
Linear Phase      : No

Implementation Cost
Number of Multipliers : 11
Number of Adders      : 10
Number of States      : 5
Multiplications per Input Sample : 11
Additions per Input Sample : 10

% -----
% Filter #2
% -----
Discrete-Time IIR Filter (real)
-----
Filter Structure   : Direct-Form II Transposed
Numerator Length  : 6
Denominator Length : 6
Stable            : Yes
Linear Phase      : No

Implementation Cost
Number of Multipliers : 11
Number of Adders      : 10
Number of States      : 5
Multiplications per Input Sample : 11
Additions per Input Sample : 10

```

Fig. 6.13: Hardware requirement for Fractional order digital differentiator

CHAPTER 7

CONCLUSION & FUTURE SCOPE

7.1 CONCLUSION

In this dissertation, digital filters have been discussed along with their time domain and frequency domain analysis. Various advantages and applications of digital filters are listed in this work. Different types of digital filters are discussed along with their characteristics.

Design methods such as impulse invariance and the bilinear transformation are explained in detail. Mathematical techniques for the design of integer order and fractional order digital differentiator have been discussed from the basics. Several techniques to design digital differentiator available in literature are also discussed.

A second order low pass IIR digital differentiator has been designed. The design of the filter is formulated as a problem of optimizing and adjustable parameter 'a'. The digital fractional order differentiator responses are obtained using MATLAB. The amount of hardware required for the various design methodologies is obtained after the simulation.

In some cases the obtained transfer functions have poles that lie on the unit circle of the z-plane which obviates the application of the stabilising method which consists of reflecting the pole that lies outside the unit circle at radius r to inside the unit circle at a radius of $1/r$ and compensating the magnitude response by multiplying the resulting transfer function by $1/r$.

The characteristics of a number of important approximations of digital differentiators and digital fractional order filters were affirmed from the results obtained. Digital differentiator designed using mathematical equations are then modeled to find the correct response by passing some test signal such as sin wave.

Magnitude and phase response of the Al-Alaoui operator and NMF model for the design of fractional order digital differentiator are discussed. The proposed low-pass differentiators were shown to have shorter transition regions, and thus better ability to suppress high frequency noise, for much lower order filters. In addition, the new low-pass differentiators exhibit almost linear phases in their corresponding passband.

7.2 FUTURE WORK

The future scope of the proposed work is as follows:

- The modelled design can be implemented on FPGA to test its performance for real time applications.
- By using different expansion techniques on the proposed mathematical model of digital differentiator optimized magnitude and phase response can be obtained.
- The approaches could well be employed to design other types of IIR filters that approximate linear phases in the passbands.
- Also the different approximations can be used to meet the magnitude specifications at a lower computational cost than the corresponding FIR filters.

PUBLICATIONS

1. Md. Tarique Siddiquee and Sanjay Kumar, “Qualitative Realization of Second Order Digital Differentiator,” *International Journal of VLSI and Embedded Systems*, Vol. 4, pp.376-378, May 2013.

REFERENCES

1. National instruments, "Introduction to Filtering," August 2012.
2. S. K. Mitra, "Digital Signal Processing", New York: Tata McGraw Hill, 2005.
3. L. Tan, and J. Jiang, "Digital Signal Processing: Fundamentals and Applications", Amsterdam: Academic Press, 2007.
4. http://en.wikipedia.org/wiki/Digital_filter
5. Application Note 779 A Basic introduction to Filters, National Semiconductor Corporation 2010.
6. M. A. Al-Alaoui, "A class of second-order integrators and low-pass differentiators," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, Vol. 42, pp. 220–223, April 1995.
7. M. A. Al-Alaoui, "Novel approach to designing digital differentiators," *IEEE Electronic Letters*, vol. 28, pp. 1376–1378, Jul. 1992.
8. B. K. Lau and V. Sreeram, "Design Of Low Order Approximately Linear Phase IIR Filters," *IEEE Symposium on Advances in Digital Filtering and Signal Processing*, pp. 92-95, June 1998.
9. I. W. Selesnick, "Maximally Flat Low-Pass Digital Differentiators," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol.49, pp. 219-223, Aug. 2002.
10. M. A. Al-Alaoui, "Class of Digital Integrators and Differentiators," *IET Signal Processing*, Vol. 5, pp. 251-260, Apr. 2011.
11. M. A. Al-Alaoui, "Novel Stable Higher Order s-to-z transforms," *IEEE Transactions on Circuits And Systems—I: Fundamental Theory And Applications*, Vol. 48, pp. 1326-1329, Nov. 2001.
12. A. Loverro, "Fractional Calculus: History, Definitions and Applications for the Engineer," Department of Aerospace and Mechanical Engineering, University of Notre Dame, May 2004.
13. M. A. Al-Alaoui, "Linear Phase Low-Pass IIR Digital Differentiators," *IEEE Transactions On Signal Processing*, Vol. 55, pp. 697- 706, Feb. 2007.

14. Y. Q. Chen and B. M. Vinagre, "A new IIR-type digital fractional order differentiator," *Signal Processing - Special issue: Fractional signal processing and applications*, Vol. 83, pp. 2359 – 2365, Nov. 2003.
15. Y. Q. Chen and K. L. Moore, "Discretization Schemes for Fractional-Order Differentiators and Integrators," *IEEE Transactions on Circuits and Systems—I: Fundamental Theory and Applications*, Vol. 49, pp. 363-367, March 2002.
16. L. Dorčák, I. Petráš, J. Terpák and M. Zborovjan, "Comparison of the Methods for Discrete Approximation of the Fractional-Order Operator," *ICCC*, pp. 851-856, May. 2003.
17. C. C. Tseng, "Design of FIR and IIR fractional order Simpson digital integrators," *Signal Processing*, Vol. 87, pp. 1045–1057, May 2007.
18. B. T. Krishna and S. S. Rao, "On design and applications of digital differentiators" *International Conference on Advanced Computing (ICOAC)*, pp. 1 – 7, Dec. 2012.
19. D. K. Upadhyay and R. K. Singh, "Recursive wideband digital differentiator and integrator", *Electronics Letters*, Vol. 47, pp. 647 – 648, May 2011.
20. A. Sarkar and S. Sengupta, "Second-degree digital differentiator-based power system frequency estimation under non-sinusoidal conditions," *Institution of Engineering and Technology Science Measurement and Technology*, Vol. 4, pp. 105– 114, May 2010.
21. D. Chen, Y. Q. Chen, and D. Xue, "Digital Fractional Order Savitzky-Golay Differentiator," *IEEE Transactions on Circuits and Systems—II: Express Briefs*, Vol. 58, pp. 758-762, Nov. 2011.
22. M. A. Al-Alaoui "Simulation and Discretization of Fractional Order Systems," *International Conference on Proceedings of the Genetic and Evolutionary Methods*, pp. 249-255, July 2009.
23. M. A. Al-Alaoui, "Al-Alaoui operator and the new transformation polynomials for discretization of analogue systems," *Springer Electrical Engineering*, Vol. 90, pp. 455–467, June 2008.
24. M. A. Al-Alaoui "Novel Class of Digital Integrators and Differentiators," *Digital Object Identifier*, 2008
25. S. W. Smith, "The Scientist and Engineer's Guide to Digital Signal Processing," *San Diego: California Technical Publications*, 1997.

26. J. O. Smith III, "Impulse-Response Representation," Introduction to Digital Filters with Audio Applications, *Stanford University Press*, August 2006.
27. B. Evans, "Digital Filters," class notes for EE 445S, Department Of Electrical And Computer Engineering, Texas University, 2013.
28. D. Rowell, "Understanding Poles and Zeros," Department of Mechanical Engineering, Massachusetts Institute of Technology, Nov. 01, 2002.
29. S. V. Ginde & J. A.N. Noronha, "Design Of IIR Filters," class notes for ECE 4624, Bradley Department of Electrical & Computer Engineering, Virginia Polytechnic and State University, Jul 1, 2002
30. Labview 2012 Manual, National Instruments, June 2012.
31. Y. Singh, S. Tripathi, M. Pandey, "Analysis of Digital IIR Filter with Lab VIEW," *International Journal of Computer Applications* , Volume 10– No.6, pp. 23-30, Nov. 2010.
32. T. Pilachowski, "The Trapezoidal Rule & Simpson's Rule," class notes for calculus
33. F. Leulmi, Y. Ferdi, "An Improvement of the Rational Approximation of the Fractional Operator s^a ," *Saudi International Conference on Electronics, Communications and Photonics*, pp.1-6, April 2011.