TRABALHO DE GRADUAÇÃO

Nonlinear Quantizer Design Based on Clenshaw-Curtis Quadrature

Bruno Rezende da Costa

Brasília, Julho de 2019

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASILIA
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

Nonlinear Quantizer Design
Based on Clenshaw-Curtis Quadrature

Bruno Rezende da Costa

Relatório submetido ao Departamento de Engenharia Elétrica como requisito parcial para obtenção do grau de Bacharel em Engenharia Elétrica

Banca Examinadora

Prof. José Edil G. Medeiros, ENE/UnB ____________________
Orientador

Prof. Leonardo R. A. X. de Menezes, ENE/UnB ____________________
Examinador Interno

Prof. Sandro A. P. Haddad, Faculdade do Gama/UnB ____________________
Examinador Externo

Dedicatória

Dedico este trabalho à minha família: meu pai, Alexandre; minha mãe, Teresa; e ao meu irmão Lucas. Por sempre se mostrarem presentes como família e amigos.

Bruno Rezende da Costa

Agradecimentos

Como não poderia deixar de ser, agradeço primeiramente à minha família, que me providenciou sempre do bom e do melhor, não somente em termos de bens e conforto, mas amor e educação. Meus pais e meu irmão certamente são os maiores responsáveis por eu ter sido capaz de realizar tamanha caminhada em minha vida. Todos os momentos de broncas, brigas, conselhos, piadas fazem da nossa família e relação unicas.

Em segundo lugar, gostaria de agradecer aos meus amigos, pelo companheirismo nas horas mais difíceis, pelos conselhos e às vezes também somente pela distração. Como seria impossível nomear todos os amigos que foram importantes ao longo de toda minha caminhada acadêmica, me aterei a citar os que estavam por perto nos últimos (e mais conturbados) meses da elaboração deste projeto.

Gostaria de agradecer à Erika, pelos momentos de carinho e pela confiança em meu pontecial. Adradeço também à Giovanna e Anna, pelas diversas madrugadas de companhia, café e estudos. Também o faço aos meus amigos Vinicius, George e Arthur, que trilharam a mesma árdua caminhada um pouco à frente de mim e foram me aconselhando e dizendo que "vai dar tudo certo". Agradeço aos meus colegas de curso, em especial aos colegas da minha turma do 2/2013 (ao Tea Bags, claramente), que fizeram da minha graduação uma experiência de fato inesquecível.

E por último gostaria de agradecer ao meu orientador, Professor Edil, pela paciência e confiança postas em mim e no meu potencial, pela franqueza e precisão nos comentários, críticas e elogios até. Em especial, agradeço pela oportunidade de dar continuidade a um trabalho de tantos anos e de tamanha relevância pessoal.

Aos demais não citados nominalmente, agradeço-os desde já por tudo que fizeram por mim, mesmo sem saber que o fizeram.

Bruno Rezende da Costa

ABSTRACT

This thesis aims to provide a novel method for designing nonlinear moment preserving quantizers based on the Clenshaw-Curtis quadrature. The basic concepts of Analog-to-Digital Converters (ADCs) are defined for contextualization of the discussed problem and to serve as a basis for understanding quantizers parameters. Then, a formal definition of the Unscented Transform (UT) is proposed for this work's context, and the key concepts of quadrature are applied to it as a mathematical tool for UT calculation. Finally, the design method is detailed, presenting the relationship between quadrature's nodes and weights and the quantizers parameters. This design is applied to a case study simulation, for validation of theoretical calculations.

RESUMO

Esta tese visa propor um novo método para projeto de quantizadores não lineares conservadores de momentos estatísticos, baseado na quadratura de Clenshaw-Curtis. Os conceitos básicos de Conversores Analógico Digital são definidos para contextualização do problema discutido e para servir de base para o entendimento dos parâmetros de quantizadores. Então, uma definição formal da Transformada da Incerteza - Unscented Transform (UT) - é proposta para o contexto deste trabalho, e os conceitos básicos de quadratura são aplicados como uma ferramenta matemática para cálculo da UT. Finalmente, a metodologia de projeto do quantizador é detalhada, apresentando a relação entre os nós e pesos de uma quadratura com os parâmetros de quantizadores. O projeto é então aplicado a uma simulação de estudo de caso para verificação dos cálculos teóricos.

Summary

List of symbols

Greek symbols

Δ	Quantization Step
Δ_{qe}	Quantization Error/Distortion
ζ	Experimental outcome
$\pi_j(x)$	Polynomial of degree j in x
$d\lambda$	Measure of a quadrature integral
λ_i	i-th quadrature weight (or Cotes Number for the Newton-Cotes case)

Variables

$f_{nyquist}$	Nyquist Sample Rate
L	ADC's number of different levels
th_n	n-th threshold level
s_i	i-th sigma-point
w_i	i-th weight
$\emptyset$	Empty set or Impossible event
$\mathcal{S}$	Space set or Certain event
x	Real Variable
$\boldsymbol{x}$	Random Variable
$\boldsymbol{x_q}$	Quantized/Digital Random Variable
m_n	n-th order statistical moment
$p_{\boldsymbol{x}}(x)$	Continuous PDF of a continuous RV $\boldsymbol{x}$
$p_{\boldsymbol{x_q}}(x_q)$	Discrete PDF of a quantized/digital RV $\boldsymbol{x_q}$
S_n	n-point partition of an interval
x_i	i-th quadrature node
$R_n(f)$	Associated approximation error of an n-point quadrature rule
$\mathbb{P}$	Space denoted by all real polynomials
$\mathbb{P}_d$	Space denoted by all real polynomials of degree at most d
$\mathtt{l}_i(x)$	i-th fundamental polynomial of the Lagrange interpolation formula

Acronyms

ADC	Analog-to-Digital Converter/Conversion
DAC	Digital-to-Analog Converter/Conversion
INL	Integral nonlinearity
DNL	Differential nonlinearity
SNR	Signal-to-noise ratio
SINAD	Signal-to-noise and distortion
SFDR	Signal-free dynamic range
FOM	Figure of Merit
PDF	Probability Density Function
CDF	Cumulative Density Function
RV	Random Variable
UKF	Unscented Kalman Filter
EKF	Extended Kalman Filter
UT	Unscented Transform
CC	Clenshaw-Curtis
FFT	Fast Fourier Transform
MATLAB	Registered Trademark of MathWorks, Inc.
MPR	Moment Preserving Ratio

Introduction

In the context of the modern digital world, one can be assured that computers play a major role in facilitating people's lives. In fact, computers act in a myriad of different contexts, from bringing the comfort of calling and texting anyone with another cellphone device in the world, to launching rockets to space, expanding the frontiers of human knowledge of the universe.

Every time a computer has to interact with the real world, it does that through sensors (of light, sound, temperature, pressure, etc). All these sensors will capture information in a format that we call Analog signal, which is a mathematical abstraction of real life events. However, the computers do not understand these kinds of information, they only work with bits ("zeros" and "ones", "yes" and "no"), and this we call Digital signals or digital information.

Therefore, one can already notice the always present need of a device that converts from one kind of information to another. Those are called Analog-to-Digital and Digital-to-Analog Converters (ADCs and DACs, respectively). These devices exist in the context of technology since the dawn of the first computers in the early 20^{th} century and have been through a series of modifications and updates. Yet, even with all these different and complex advances, the ADC can be divided in three major processes: the sampling phase, the quantization phase and the coding phase. All these processes are going to be better and formally detailed in Chapter 1. This thesis, however, focus its attention on the quantization process.

Motivation and Problem Statement

The quantization is an essential process for any kind of ADC. But in fact, most of the quantizers used (the linear quantizers) are not optimized for the extremely complex and precise operations in which they are required. Complex operations generally present extremely nonlinear behaviours, context in which linear quantizers, by definition, fail to present optimal results.

Moreover, quantization is a delicate process of the conversion for yet another reason. Differently than the other two parts, it always inserts errors/distortions into the system. That is, we necessarily have some loss of information every time we try to implement an ADC, because of the quantization process, and the reasons for that are going to be better explained along Chapter 1.

Proposed Solution

This work thrives to tackle the presented problems by proposing a novel design for nonlinear quantizers. For that, we will work on the intersection of apparently unrelated fields, such as ADCs, Moment Preserving transformations and Numerical Quadratures.

The design itself consists in a generic methodology for calculating the quantizer's constructive parameters (its output levels and input thresholds) for arbitrary signals, but that at the same time adapts itself to different input behaviours. This sounds contradictory, but it only means that the method for calculating the parameters does not have to change based on the signal input, consequently the architecture itself is the same for every signal, but it actually collects information from the input and calculates the quantizers parameters based on those measures, adapting its characteristic curve (thus its general behaviour) for different signals.

With that abstract idea in mind, let us introduce the specifics of the methodology being proposed. The information measure for the signal, for instance, is not going to be directly its amplitude, but the input signal's statistical moments. It can be proven that a signal's mean and variance measures (the first and second moments) carry information about the signal itself, the same way as the other higher moments. Therefore, if we inted to solve the fact that quantization is an error inducing process, then we need to find a way of preserving the signal's moments during the conversion process.

The idea of preserving statistical moments leads to an important mathematical framework named the Unscented Transformation (UT). It was an idea proposed in 1997 in the context of control systems designs to avoid approximating complex systems through their linearization. This mathematical framework proposed to discretizate a continuous signal, in order to diminish the calculations complexity. As one can already observe, this is exactly what we are aiming for when dealing with conversion processes of highly nonlinear signals, therefore this transformation comes quite in hand for reasons we will better clarify during Chapter 2.

The UT has many different algorithms implementations in the context of control systems, but they tend to be very specific for this kind of problem. We then propose a different approach that was very little explored in digital processing chains context throughout the last decade or more. This approach consists of using some mathematical tools known as Numerical Quadratures, by means of orthogonal polynomials use. These apparently unrelated field are going to be connected to the UT context in Chapter 3.

And finally, with all these tools at disposition, we can develop the design of a quantizer, in which its quantization levels (or output levels) are related to the nodes of a quadrature calculation, and the input thresholds are related to the quadrature's weights. This terms and what they mean are going to be formally defined along this thesis and in Chapter 4 a refined explanation of the design proposed is going to be given.

Objectives

To sum it all up, we can state that the main objectives of this thesis are:

1. Propose and validate a novel methodology for designing signal-generic nonlinear quantizers based on Clenshaw-Curtis quadrature methods;
2. Verify in simulation environment if the designed quantizers present better results than the linear one;

We follow by presenting a brief introduction on the Electronics of ADCs.

Chapter 1

Topics in Analog to Digital Converter

This chapter is the outer layer of this thesis and it is intended to introduce some basic concepts of Analog-to-Digital Converters (ADCs) for a better understanding of the main application of the presented methodology. First, we start by defining the main processes of the ADC, which are the sampling, quantization and coding, Figure 1.1 shows a block diagram which illustrates well the signal flow during the conversion process. After that, we present some of the main metrics used to characterize ADCs functionality and a new metric proposed for analyzing specifically nonlinear moment preserving converters.

1.1 Sampling

Any analog signal is an abstraction of natural events, i.e., they are a mathematical representation of what happens in the real world. Such a representation many times takes the form of a function $f_{analog}(t)$, such that $f_{analog} : \mathbb{R} \mapsto \mathbb{R}$ and the Domain t of this function is the time. That being said, one can observe that an analog signal is continuous on both its Domain and Codomain, which means that it can assume any value (of amplitude, for example) on any given time. Sampling

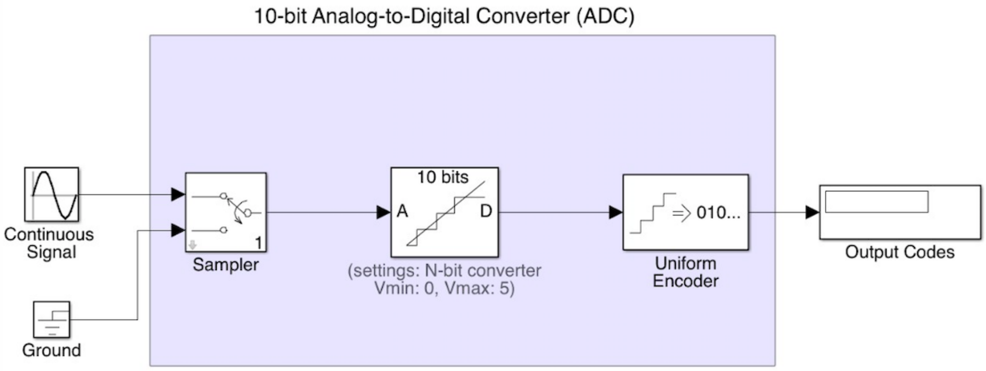

10-bit Analog-to-Digital Converter (ADC)
Continuous Signal
Ground
Sampler
1
10 bits
A
D
(settings: N-bit converter Vmin: 0, Vmax: 5)
010...
Uniform Encoder
Output Codes

Figure 1.1: Block Diagram for general ADC showing sampling, quantization and coding

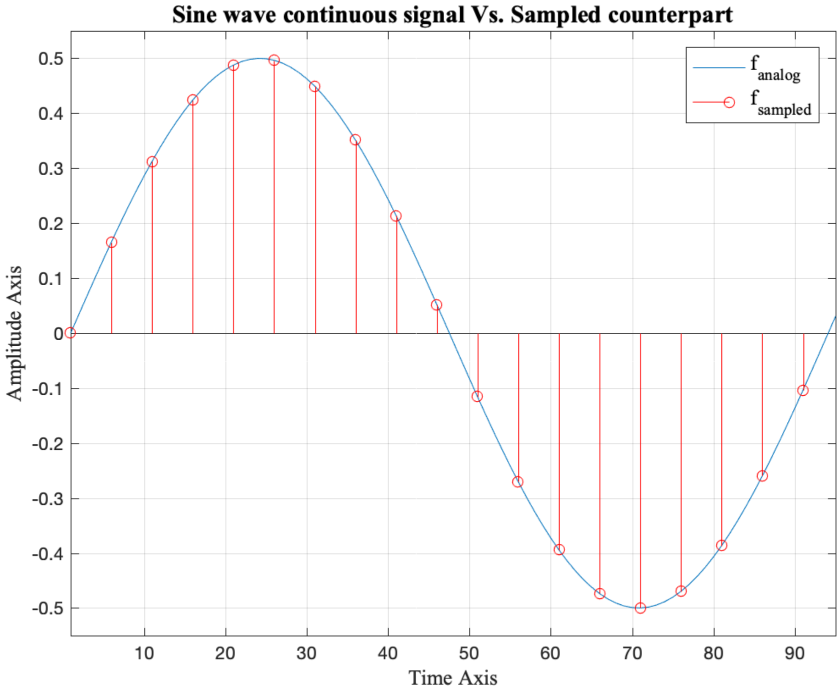


Figure 1.2: Continuous Signal and its sampled counterpart

is the process responsible for the discretization of the Domain of this analog signal, transforming the former $f_{analog}(t)$ into a

$$f_{sampled} : \mathbb{Z} \mapsto \mathbb{R},$$

such that, $f_{sampled}[n] = f(n \cdot T)$, where $n \in \mathbb{Z}$ and T is the *sample period.* This means that the new function $f_{sampled}$ assumes the same values as the former f_{analog}, but it is not defined for every value of time, it is only defined for integer amounts of periods T.

One would be led to believe that there is loss of information after this process, since we no longer have the information of this signal for any given time, but only for a finite amount of *samples.* However, based on the theory of Fourier analysis, Nyquist and Shannon proved that if the sample time $T \leq \frac{1}{2B}$, where B is the highest frequency of the signal spectrum, the analog signal f_{analog} can be completely reconstructed from its discrete-time counterpart $f_{sampled}$ [1][2]. This is know as the Nyquist Criterion. However, as we will see on the following section, an analog criterion does not exist for the quantization process, which causes information losses in every case.

In practice, however, this information preserving sampling process can never be achieved, since baseband signals with a finite spectrum (also know as band-limited signals) do not exist in nature, neither does the ideal sampler (which would be the impulse train, referred in [3]). We can nonetheless achieve an almost distortion-free sampling process considering the use of oversampling[1] and pre-processing techniques such as anti-aliasing filters, high order Low Pass Reconstruction Filters and Equalizers, but those techniques are also out of the scope of this work.

[1] usually it is the use of a sample rate 5 times the Nyquist Sample Rate ($f_{nyquist} = 2B$)

1.2 Quantization

If the sampling is responsible for discretizing the Domain of f_{analog}, the quantization is the one responsible for the discretization of its Codomain. This process limits the value possibilities of f_{analog} from a range of continuous real values to a finite number of possible values by "rounding off" those real values to the nearest so called *quantization level.* This approximation is what causes the quantization to always insert error in the system (also called *quantization error* or *distortion* Δ_{qe}) and it is the main reason why this process should be given the proper attention, in order to mitigate this distortion as much as possible.

As shown in Figure 1.1, normally the quantization is represented as the process that follows the sampling. That being said, we can define a function $f_{digital}$, which is the resulting function that represents a signal sampled and quantized:

$$f_{digital} : \mathbb{Z} \mapsto \mathbb{Z}.$$

The error associated with this process is defined in terms of the number of different *quantization levels* of a given ADC, which is known as the ADC's resolution. For that, we will define $L = 2^N$, given that N is the number of bits of the ADC and L as the number of its different levels. In a linear quantizer, those levels are equally distributed over a certain interval $[X_{Q-min}, X_{Q-max}]$[2]with a step of $\Delta = \frac{X_{Q-max}-X_{Q-min}}{L}$, such that the threshold levels th_n are defined as:

$$th_n = X_{Q-min} + n \cdot \Delta, \quad \{n = 0, 1, 2, ..., L\} \tag{1.1}$$

and the values to which the analog inputs $f_{sampled}$ are going to be mapped to are defined by:

$$f_{digital}[n] = \frac{th_n + th_{n-1}}{2} \tag{1.2}$$

This means that every input value of $f_{sampled}[n] \in [th_{n-1}, th_n]$ is going to be mapped to an output quantization level $f_{digital}[n]$, defined by equation 1.2. The error itself is defined as[3]:

$$\Delta_{qe} = f_{sampled}[n] - f_{digital}[n] \tag{1.3}$$

However, these definitions presented by equations 1.1 and 1.2 are only valid for uniform quantizers. For nonuniform (or nonlinear) quantizers this input-output mapping can be done by means of any arbitrary function. For instance, in biomedical applications a logarithmic function is often used instead of a simple mean value between the threshold levels to define the quantizer behaviour (see [4],[5] and [6] for research examples in this area). As we will deeply explain in the following sections, this work proposes a trigonometric distribution of threshold levels (specifically, a distribution equal to the Chebyshev nodes), and a mapping definition based on the Moment-Preserving UT.

[2] X_{Q-max} and X_{Q-min} are parameters of the quantizer, not of the signal; If $max(f_{analog}) > X_{Q-max}$ or $min(f_{analog}) < X_{Q-min}$, the quantizer is said to be overloaded (or saturated).

[3] Note that this definition for the Δ_{qe} is independent of whether or not the quantizer is linear

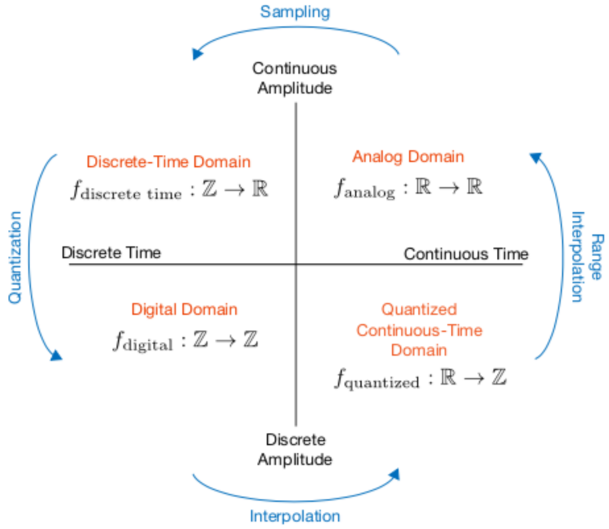

Sampling
Continuous Amplitude
Discrete-Time Domain
$f_{\text{discrete time}} : \mathbb{Z} \to \mathbb{R}$
Analog Domain
$f_{\text{analog}} : \mathbb{R} \to \mathbb{R}$
Quantization
Range Interpolation
Discrete Time
Continuous Time
Digital Domain
$f_{\text{digital}} : \mathbb{Z} \to \mathbb{Z}$
Quantized Continuous-Time Domain
$f_{\text{quantized}} : \mathbb{R} \to \mathbb{Z}$
Discrete Amplitude
Interpolation

Figure 1.3: Signal Domains and Definitions
Source: Image taken from [7], with author's consent

To sum it up, we will refer to the Figure 1.3, in order to illustrate the context in which each signal is defined and what they represent for a better comprehension of this and prequel sections. (Note that $f_{discrete\ time}$ is another nomenclature for the defined $f_{sampled}$ function.)

1.3 Coding

The coding process is the association of every quantization level to a different symbol (normally a group of n binary digits) which is going to be transmitted via telecommunication systems or processed in a CPU. The evolution of studies and techniques in this process became another good reason why the digital systems started to take over the place of analog applications. These techniques help to detect or even correct distortions in the signal via redundancy, cyclic, convolutional and Hamming codes. For a far more in-depth discussion refer to [3].

1.4 ADC's Metrics

Here in this section it is going to be presented a couple standard metrics used when analyzing general ADC's functionality. For a more detailed explanation, refer to [8], [9] or manufacturer's data sheets. Figures 1.4 and 1.5 illustrate some of the following definitions:

- **Static Parameters:**
  - **Differential nonlinearity (DNL)** is the maximum deviation from one output level to the next. The ideal value would be of one least significant bit (LSB), which is the further most bit to the right.

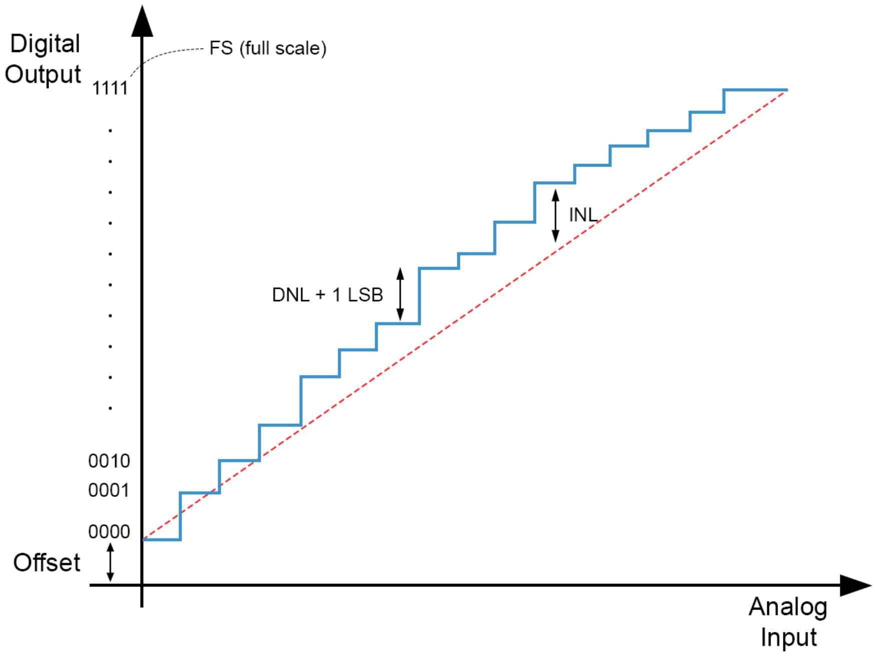


Figure 1.4: Representation of ADC's Static Parameters

- **Integral nonlinearity (INL)** is the maximum deviation of the so called ADC's *input/output characteristic* from the reference straight line (represented in the Figure 1.4 as the dashed diagonal line which passes through the characteristic's end points). The difference between the ideal and actual characteristics will be called the INL profile.
- **Offset** is where the ADC's characteristics curve actually intercepts the vertical line (the Digital Output axis in the example's case).
- **Gain error** is the deviation of the slope of the reference (dashed) line from its ideal value (usually unity).

- **Dynamic Parameters:**[4]

  - **Latency** is the total delay from the time the input changes to the time the output has settled within a specified value (inside a threshold band around its final value).
  - **Signal-to-noise ratio (SNR)** is the ratio of the signal power to the noise power signal, normally modeled as an Additive-White-Gaussian Noise or just AWGN (see [3] for a more detailed explanation on this).
  - **Signal-to-noise and distortion ratio (SINAD)** is the ratio of the signal power to the total noise plus harmonic distortions at the output. In this work's application this distortions are mostly caused by the quantization, then called *quantization spurs*. This ratio is also sometimes referred as Signal-to-(noise+distortion) ratio (SNDR). See Figure 1.5 for a graphic representation of the difference between this and SNR ratios.

- **Spurious-free Dynamic Range (SFDR)** is ratio of the signal power to the maximum noise/distortion power in the output. In Figure 1.5 it is shown as the region between the signal's peak and the noise/distortion peak (it is in fact the difference between these two values for the graphic's scale is in dB, which results in a ratio between the absolute power values in Watts).

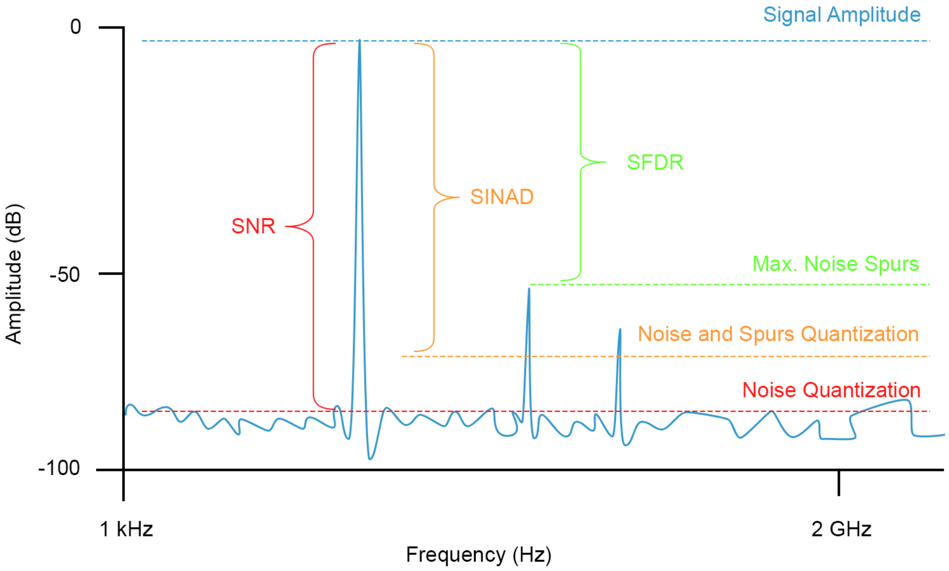


Figure 1.5: Representation of ADC's Dynamic Parameters

All of these parameters are widely used as means of measuring ADC's and DAC's real behaviour and their deviation from theoretical/ideal behaviour. However, all of those consider that the converter being analyzed is a linear one, which is not the case sometimes (including the one being proposed by this thesis). As already stated in the works of Santos et.al [10], a different *figure of merit* (FOM) has to be proposed as a mean to analyze this nonlinear behaviour not as a flaw but as an expected/desired one. That is, the characteristic curve of a nonlinear converter is not going to be a straight line (by definition), but there has to be a way of measuring deviations from this expected nonlinear curve.

We propose such a FOM on Chapter 4 based on the conservation of statistical moments of the input signal. Since the converter is designed to conserve the input signal's higher order moments, the output signal's moments can be calculated and a relative error used as a deviation parameter. The actual use of this parameter is going to be clarified in the simulations chapter, where the designed ADC's behaviour is going to be analyzed via this method.

[4] there are other standard Dynamic Parameters like *Glitch impulse area* and *Settling Time*, which are commonly used to analyze linear DAC's behaviour as well.

Chapter 2

Topics in Unscented Transform

In the previous chapter we discussed about the basic concepts of an ADC, its main processes and metrics used to evaluate its performance. This was all to give the context in which the rest of the thesis is going to be applied, whereas this chapter aims to provide some fundamental mathematical framework that is going to be used for the nonlinear quantizer design. This consists on the presentation of the Unscented Transform.

For a more complete comprehension on each of these topics, we provide some background on basic Probability and Statistics concepts which will later turn to be useful. Moreover, we present a definition for the UT and why is it of interest for the context in which this thesis is inserted. Then, we present an example case of UT application for a better understanding of the given definitions.

2.1 Probability Theory

The probability theory deals with average of occurring events. The purpose of this theory is to describe and predict such averages in terms of probabilities of events [11]. Based on the Set Theory, we will expose the following definitions.

Definition 1 *Given an experiment preformed n times, the event $\mathcal{A}$ is a subset of $\mathcal{S}$[1] which includes a number of experimental outcomes ζ_i that occur $n_{\mathcal{A}}$ times. Then, for n sufficiently large, we can say that $P(\mathcal{A})$ (the probability of occurrence of the event $\mathcal{A}$) is associated with the relative frequency of occurrence of $\mathcal{A}$. In other words:*

$$P(\mathcal{A}) = \frac{n_{\mathcal{A}}}{n}, \tag{2.1}$$

for n large enough.

The next definitions are actually axioms that defines the probability theory as a mathematical representation of physical phenomena.

Definition 2 *Let $\mathcal{A}$ and $\mathcal{B}$ be any two events of the space $\mathcal{S}$, then:*

$$P(\mathcal{A}) \geq 0 \quad \textit{and} \quad P(\mathcal{B}) \geq 0 \tag{2.2a}$$

[1] $\mathcal{S}$ is called the *certain event*

$$P(\mathcal{S}) = 1 \tag{2.2b}$$

$$\textit{If} \quad \mathcal{A} \cap \mathcal{B} = \varnothing \quad \textit{then} \quad P(\mathcal{A} \cup \mathcal{B}) = P(\mathcal{A}) + P(\mathcal{B}) \tag{2.2c}$$

As a consequence of these 3 axioms, it can be shown also that the probability of the empty set ($\emptyset$, also called the *impossible event*) is zero:

$$P(\emptyset) = 0$$

There is another important condition, besides Axioms 2.2a - 2.2c, that determine the set of properties obeyed by all probabilities, which is:

Definition 3 *Given that the class of all subsets of $\mathcal{S}$ is a Borel Field*[2]*, if we consider infinitely many subsets $\mathcal{A}_i$ of $\mathcal{S}$, such that $\mathcal{A}_1 \cap \mathcal{A}_2 \cap ... = \varnothing$, then:*

$$P(\mathcal{A}_1 \cap \mathcal{A}_2 \cap ...) = P(\mathcal{A}_1) + P(\mathcal{A}_2) + ... \tag{2.3}$$

This last definition is important to determine probabilities not only to a finite union and intersection of set, but also to their limits [11] (which will be important later on when we talk about probabilities of continuous random variables). The condition determined by 2.3 is known as the *axiom of infinite additivity.*

In the context of this thesis, it is important to define probabilities and events for a set $\mathcal{S}$ as the set of all real numbers. It can be shown that it is impossible to assign probabilities to elementary events[3]of $\mathcal{S}$ that satisfy all axioms defined by 2.2a - 2.2c and 2.3. For that, we will define an event as being a set $\{x \mid x_1 \leq x \leq x_2\}$, given that x_1 and x_2 are any real number [11].

2.1.1 Random Variables, Cumulative and Density Functions

Definition 4 *A Random Variable (RV) $\boldsymbol{x}$ is an arbitrary function which maps every outcome ζ of an experiment to a number $\boldsymbol{x}(\zeta)$. These functions satisfies only two conditions:*

I The set $\{\boldsymbol{x} \leq x\}$ for every x;

II The probabilities of the events $\{\boldsymbol{x} = \infty\}$ and $\{\boldsymbol{x} = -\infty\}$ are equal to zero, i.e.,

$$P\{\boldsymbol{x} = \infty\} = 0 \quad \textit{and} \quad P\{\boldsymbol{x} = -\infty\} = 0$$

RVs can be defined for any real or complex values, but in this whole thesis we will consider only $x \in \mathbb{R}$, or in other terms $x \in (-\infty, \infty)$. As such, the probabilities of the events in the extremes of the Domain of x are considered impossible.

[2] A *Borel Field*, in a short explanation, is an algebraic *field* that can be infinitely partitioned. For more details on that, refer to [11] or [12].

[3] Those are events that contain a single experimental outcome, i.e., an event $\mathcal{A} = \{\zeta_i\}$. The probability of such an event is denoted by $P\{\zeta_i\} = p_i \geq 0$

Definition 5 *The Cumulative Distribution Function (CDF) of an RV $\boldsymbol{x}$ is a monotonic non-decreasing function ($F_{\boldsymbol{x}}(x) \geq 0, \forall x \in \mathbb{R}$) defined by the following equation:*

$$F_{\boldsymbol{x}}(x) = P\{\boldsymbol{x} \leq x\}, \tag{2.4}$$

which represents the probability of the RV x This function fully describes the distribution of probability of its associated RV. Also, note that, by definition (and based on the probability's axioms), we can define the following two equalities:

$$\lim_{x \to -\infty} F_{\boldsymbol{x}}(x) = 0 \quad \textit{and} \quad \lim_{x \to \infty} F_{\boldsymbol{x}}(x) = 1 \tag{2.5}$$

For disambiguation purposes, note that x is any real number, whereas $\boldsymbol{x}$ represents the RV which maps an event to this number. This convention will be maintained throughout the rest of the thesis.

Definition 6 *The Probability Density Function (PDF) of an RV $\boldsymbol{x}$ is defined in terms of the CDF associated to the same RV, such that:*

$$p_{\boldsymbol{x}}(x) = \frac{dF_{\boldsymbol{x}}(x)}{dx}, \tag{2.6a}$$

for every $x \in (-\infty, \infty)$. It is also possible to define the PDF in integral terms, such as:

$$\int_{-\infty}^{x} p_{\boldsymbol{x}}(\tau) d\tau = F_{\boldsymbol{x}}(x). \tag{2.6b}$$

It is also important to state an important condition (that can be derived from the axioms 2.2a-2.2c) for the PDF to be meaningful in the context of probability, which is the fact that $\int_{-\infty}^{\infty} p_{\boldsymbol{x}}(x)dx = 1$, always.

2.1.2 Continuous and Discrete Distributions

Based on these definitions, we can now define important RV types, such as *continuous* and *discrete* distributions.

Definition 7 *An RV is called continuous if and only if its CDF is continuous, i.e.,*

$$F_{\boldsymbol{x}}(x^+) \;=\; F_{\boldsymbol{x}}(x^-) \;=\; F_{\boldsymbol{x}}(x), \; \textit{for any real } x,$$

where $F_{\boldsymbol{x}}(x^+)$ and $F_{\boldsymbol{x}}(x^-)$ are defined as:

$$F_{\boldsymbol{x}}(x^+) = \lim_{\epsilon \to 0^+} P\{\boldsymbol{x} \leq x + \epsilon\} \quad \textit{and} \quad F_{\boldsymbol{x}}(x^-) = \lim_{\epsilon \to 0^+} P\{\boldsymbol{x} \leq x - \epsilon\}$$

Thus, we can show that:

$$P\{\boldsymbol{x} = x\} = 0, \quad \forall x \in \mathbb{R}$$

Definition 8 *An RV is called discrete if and only if its CDF follows a staircase pattern as shown in Figure 2.1, thus presenting at least one point x_i of discontinuity, such that $F_{\boldsymbol{x}}(x_i^+) \neq F_{\boldsymbol{x}}(x_i^-)$. Hence, we can show that:*

$$F_{\boldsymbol{x}}(x_i) - F_{\boldsymbol{x}}(x_i^-) = P\{\boldsymbol{x} = x_i\} = p_i$$

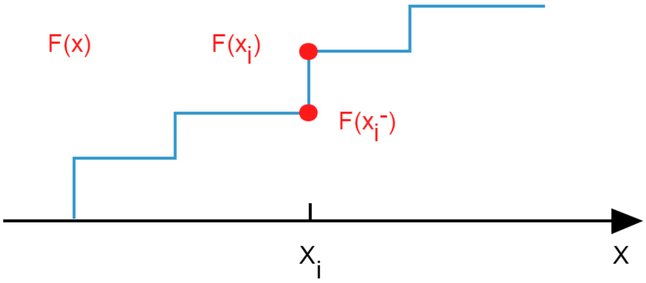


Figure 2.1: Staircase pattern for discrete RV CDFs

2.1.3 Statistical Moments

We will follow by defining the *expected value* or *mean value* for both discrete and continuous RVs, and then generalize these definitions for high order moments.

Definition 9 *The mean value of a continuous RV $\boldsymbol{x}$ is defined in terms of the expectation (operator) E(.) of $\boldsymbol{x}$:*

$$E\{\boldsymbol{x}\} = \int_{-\infty}^{\infty} x p_x(x) dx \,. \tag{2.7}$$

This is also known as the First Moment of the RV $\boldsymbol{x}$ for a given density distribution $p_{\boldsymbol{x}}(x)$.

Definition 10 *The mean value of a discrete RV $\boldsymbol{x}$ is defined in terms of its elementary probabilities $P\{\boldsymbol{x} = x_i\} = p_i$ and the operator E(.):*

$$E\{\boldsymbol{x}\} = \sum_i x_i p_i \,. \tag{2.8}$$

This definition can also be mathematically deduced from Definition 8 and 9 (for more details on that refer to [11]).

As a generalization of the previous two definitions we can now define higher order moments.

Definition 11 *The quantities:*

$$m_n = E\{\boldsymbol{x}^n\} \tag{2.9}$$

are defined as the n-th order moment of an RV $\boldsymbol{x}$, such that:

$$E\{\boldsymbol{x}^n\} = \int_{-\infty}^{\infty} x^n p_{\boldsymbol{x}}(x) dx \quad \textit{for } \boldsymbol{x} \textit{ continuous}, \tag{2.10a}$$

$$E\{\boldsymbol{x}^n\} = \sum_i x_i^n p_i \quad \textit{for } \boldsymbol{x} \textit{ discrete} \tag{2.10b}$$

2.2 The Unscented Transform

In this section, it will be presented the main theory which motivated the reasoning used to tackle the problems mentioned at the Introduction Chapter, specifically presenting the framework with which we are going to work: the Unscented Transform.

In the context of control systems, the researches done by Julier and Uhlmann became more and more relevant in the academical community, to the point that one of their most famous works [13] has more than a thousand citations and eight thousand reads[4]. They presented various implementations of nonlinear filtering methods known as Unscented and Extended Kalman Filters (respectively UKF and EKF) and developed this theory as basis for a variety of different applications (refer to [14] for references on those many employments).

The UT itself was first proposed in 1997 [15] as an efficient method for computing means and covariances of transformed random vectors, which played an important role in applications of the UKFs. However, the context in which this thesis is going to be based is more related to works done by Menezes et al. [16][17], in which the UT is presented as a way to model continuous probabilities distributions by discrete ones (see Figure 2.2), while conserving its statistical moments up to any desired order.

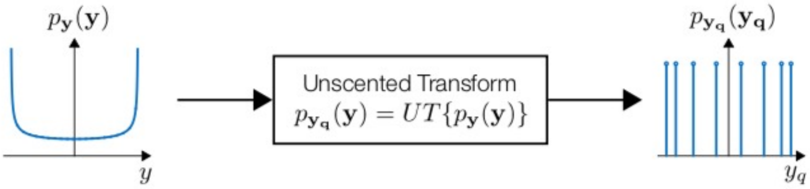

$p_{\mathbf{y}}(\mathbf{y})$
y
Unscented Transform
$p_{\mathbf{y_q}}(\mathbf{y}) = UT\{p_{\mathbf{y}}(\mathbf{y})\}$
$p_{\mathbf{y_q}}(\mathbf{y_q})$
y_q

Figure 2.2: Graphic interpretation of the UT

Source: Image taken from [7], with author's consent

This approach is justified by the fact that is easier to simulate the behaviour of an arbitrary nonlinear application over a discrete set of points than over a continuous one, and that the signal's statistical moments contain information about the signal itself. Hence, the interest in efficiently approximate the signal's PDF conserving as many moments as possible.

To intuitively justify this last statement, let $\boldsymbol{x}$ be a continuous RV with $p_{\boldsymbol{x}}(x)$ as its PDF, and $\boldsymbol{y} = g(\boldsymbol{x})$ the resulting RV of a smooth nonlinear application $g(\cdot)$ over $\boldsymbol{x}$. The Taylor Expansion of $g(\cdot)$ centered at zero results in the following series:

$$g(x) \approx \sum_{n=0}^{k} \frac{g^{(n)}(0)}{n!} x^n = \sum_{n=0}^{k} a_n x^n, \tag{2.11}$$

where $g^{(n)}(0)$ is the n-th derivative of $g(\cdot)$ evaluated at $x = 0$, and $\frac{g^{(n)}(0)}{n!} = a_n$.

Since the E(.) is a linear operator [11][12], we can apply it over the Equation 2.11, resulting in:

[4] Estimations from IEEE Xplore Digital Library. Research done by Menegaz [14] as contextualization, see its introduction footnote for link reference.

$$
\begin{aligned}
E\{g(x)\} &\approx E\{\sum_{n=0}^{k} a_n x^n\} \\
\Leftrightarrow E\{g(x)\} &\approx \sum_{n=0}^{k} a_n E\{x^n\}
\end{aligned}
\tag{2.12}
$$

Equation 2.12 shows us that knowledge of the input's moments $E\{x^n\} = m_n$ is sufficient to determine the nonlinear behaviour of $g(\cdot)$. Moreover, the bigger the k, the more moments information we have, and the closer this approximation gets to an equality. Note that Equations 2.11 and 2.12 also hold true for RV inputs and $\boldsymbol{y} = g(\boldsymbol{x})$.

Finally, we define the UT:

Definition 12 *The α-th order Unscented Transform of a given continuous PDF $p_{\boldsymbol{x}}(x)$ will be defined by the operator $UT^{\alpha}(\cdot)$, such that:*

$$UT^{\alpha}\{p_{\boldsymbol{x}}(x)\} = \{s_i, w_i\}_n, \tag{2.13}$$

where $\{s_i, w_i\}_n$ is a n-set of sigma-points and weights pairs, such that

$$E\{\boldsymbol{x}_{\boldsymbol{q}}^k\} = \sum_{i=1}^{n} s_i^k w_i = \int_{-\infty}^{\infty} x^k p_{\boldsymbol{x}}(x)dx = E\{\boldsymbol{x}^k\}, \tag{2.14}$$

where k = 0,1,2...,α; $\boldsymbol{x}_{\boldsymbol{q}}$ is a discrete RV; and $\boldsymbol{x}$ a continuous RV.

Equation 2.13 means that the UT of a given continuous PDF is defined by a n-set of sigma-points and weights pairs ($\{s_i, w_i\}_n$), which characterizes a discrete probability distribution $p_{\boldsymbol{x}_{\boldsymbol{q}}}(x_q)$. Note that Equation 2.14 clearly states that both RVs (discrete and continuous) have the same moments (up to the α-th order), and that the n-set pair of points that solves this system of equations also defines a discrete approximation $p_{\boldsymbol{x}_{\boldsymbol{q}}}(x_q)$ for an arbitrary $p_{\boldsymbol{x}}(x)$.

From a specific set of sigma-points and weights pairs, one can also define the inverse operator which would return the continuous PDF associated to it. However, this operation is not so simple and it ends up falling into a much bigger and older problem referred in the literature as the *problem of moments* which is out of the scope of this thesis (refer to [7] for a more complete presentation of this problem and [18] for in depth discussion of it).

2.2.1 Direct Method for the UT computation

The n-set of sigma-points and weights resulted from the UT application over a continuous PDF is the solution of a nonlinear system of equations that come from the Equation 2.14. In order to clarify what does that mean, let us present an example case of the method proposed by Tabatabai et al. [19] and da Costa Junior [20].[5]

[5] This example is also detailed in the works of [7].

Let there be a PDF $p_{\boldsymbol{x}}(x)$ of an RV $\boldsymbol{x}$ that is going to be approximated to a discrete PDF $p_{\boldsymbol{x_q}}(x)$ through the use of the UT. This discrete PDF is going to be defined by a set of 3 sigma-points and weights $\{(s_1, w_1), (s_2, w_2), (s_3, w_3)\}$ which satisfy the criteria imposed by Equation 2.14.

Consider also that the statistical moments of this distribution are known up to the 5^{th} order, then we have:

$$s_1^0 w_1 + s_2^0 w_2 + s_3^0 w_3 = m_0, \tag{2.15a}$$

$$s_1^1 w_1 + s_2^1 w_2 + s_3^1 w_3 = m_1, \tag{2.15b}$$

$$s_1^2 w_1 + s_2^2 w_2 + s_3^2 w_3 = m_2, \tag{2.15c}$$

$$s_1^3 w_1 + s_2^3 w_2 + s_3^3 w_3 = m_3, \tag{2.15d}$$

$$s_1^4 w_1 + s_2^4 w_2 + s_3^4 w_3 = m_4, \tag{2.15e}$$

$$s_1^5 w_1 + s_2^5 w_2 + s_3^5 w_3 = m_5, \tag{2.15f}$$

where the m_i quantities are the ones defined by Equations 2.9 and 2.10a.

Note that we have 6 equations with 6 unknown values (the sigma-points and weights). It is obviously a nonlinear set of equations, but we can use some artifacts to simplify its solution.

We can then define a 3^{rd} order polynomial $\pi_3(x)$ whose roots are the desired sigma-points:

$$\pi_3(x) = (x - s_1)(x - s_2)(x - s_3) = x^3 + a\,x^2 + b\,x + c \tag{2.16}$$

Next, we ought to construct this polynomial from equations 2.15a-2.15d by multiplying those equations by the coefficients a, b and c in order to get a set equations that form the right-hand side of Equation 2.16.

$$\begin{aligned} c \cdot (s_1^0 w_1 + s_2^0 w_2 + s_3^0 w_3) &= c \cdot m_0 \\ b \cdot (s_1^1 w_1 + s_2^1 w_2 + s_3^1 w_3) &= b \cdot m_1 \\ a \cdot (s_1^2 w_1 + s_2^2 w_2 + s_3^2 w_3) &= a \cdot m_2 \\ (s_1^3 w_1 + s_2^3 w_2 + s_3^3 w_3) &= m_3 \end{aligned}$$

By adding these four equations together and putting the common factors in evidence we have:

$$\begin{gathered} (c + b\,s_1 + a\,s_1^2 + s_1^3) \cdot w_1 + (c + b\,s_2 + a\,s_2^2 + s_2^3) \cdot w_2 + (c + b\,s_3 + a\,s_3^2 + s_3^3) \cdot w_3 = \\ \pi_3(s_1) \cdot w_1 + \pi_3(s_2) \cdot w_2 + \pi_3(s_3) \cdot w_3 = \\ c\,m_0 + b\,m_1 + a\,m_2 + m_3 = 0, \end{gathered}$$

because, by construction the $\pi_3(x)$ polynomial has its zeros located in the s_i values.

Repeating the same process over the rest of the equations of 2.15 in groups of four equations at a time produces the following result:

$$\begin{aligned} m_3 + a\,m_2 + b\,m_1 + c\,m_0 &= 0 \\ m_4 + a\,m_3 + b\,m_2 + c\,m_1 &= 0 \\ m_5 + a\,m_4 + b\,m_3 + c\,m_2 &= 0 \end{aligned}$$

In matrix form, it would be:

$$\begin{bmatrix} m_2 & m_1 & m_0 \\ m_3 & m_2 & m_1 \\ m_4 & m_3 & m_2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} -m_3 \\ -m_4 \\ -m_5 \end{bmatrix} \tag{2.17}$$

The solution of this system of equations will give us the coefficients a, b and c. With those the polynomial $\pi_3(x)$ in 2.16 can be entirely defined, so as its roots, which are by construction equal to the desired sigma-points s_i.

The next step is to get back to the Equations 2.15a-2.15c, substitute the s_i for the calculated roots of $\pi_3(x)$ and solve the equations for weights w_i:

$$\begin{aligned} w_1 + w_2 + w_3 &= m_0 \\ s_1 w_1 + s_2 w_2 + s_3 w_3 &= m_1 \\ s_1^2 w_1 + s_2^2 w_2 + s_3^2 w_3 &= m_2 \end{aligned}$$

which in matrix form become:

$$\begin{bmatrix} 1 & 1 & 1 \\ s_1 & s_2 & s_3 \\ s_1^2 & s_2^2 & s_3^2 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} m_0 \\ m_1 \\ m_2 \end{bmatrix} \tag{2.18}$$

This completely solves the problem of calculating the pairs of sigma-points and weights for a given set of statistical moments. This approach has some advantages and disadvantages. One advantage is the fact that one can use this method for calculating the UT of a given signal input even without knowing its probability distribution function, only the statistical moments knowledge suffice. The big disadvantage is that the problem presented by Equation 2.17 is as ill-conditioned as the problem of Hilbert matrices inversion, as shown by Gautschi [21]. This fact causes instability issues during numerical computation of the solution of the referred systems of equation which makes this method impracticable for real applications.

With that in mind, this thesis proposes a slightly different approach for the Direct Method computation, which achieves a turn around this matrix problem. The Modified Direct Method approach is going to be presented in Chapter 4 and uses the Numerical Quadrature theory, more specifically the Clenshaw-Curtis technique for quadrature calculations to achieve the proposed results. Those mathematical tools are going to be presented in the following chapter.

Chapter 3

Topics in Numerical Quadrature

At this point we already presented the main context in which this thesis is inserted (the ADC topics), that represent the outer layer of this work, and in the previous chapter we presented some important and more specific frameworks and concepts to begin developing ideas related to the specific design proposed. Now, we present some mathematical background on the concepts involving Numerical Quadratures and how this and the UT (two apparently unrelated fields) intersect.

We will start by presenting the basic concepts on Mechanical and Interpolatory Quadratures (based on the works of [22] and [21]), then it will already be possible to understand how these fields intersect. Afterwards, we will define what are Orthogonal Polynomials and present one important theorem with which we will be able to use the quadrature theory presented for some specific UT calculations (also used in the works of Medeiros [7], for the same purpose). We then quickly present some comparison between the CC technique and the classical Gaussian Quadrature, to justify the use of the previous in lieu of the last (see [23]).

3.1 Mechanical Quadratures

Consider a finite or infinite interval $[a, b]$, such that S_n is a partition of this interval defined as:

$$S_n : a \leq x_1 < ... < x_n \leq b.$$

We can define the Mechanical Quadrature as the problem of numerically calculating integrals by determining a finite sum of *nodes* and *weights*.

Definition 13 *An n-point Quadrature rule is defined by the following formula:*

$$\int_{\mathbb{R}} f(x) d\lambda(x) = \sum_{i=1}^{n} \lambda_i f(x_i) + R_n(f), \tag{3.1}$$

where $d\lambda(x)$ is the measure of the approximated integral with respect to x, λ_i are the weights, x_i are the nodes (the points in which the function f is going to be sampled) and R_n is the associated error of this approximation process.

Definition 14 *A Quadrature rule is said to have a degree of exactness d if:*

$$R_n(p) = 0, \forall p \in \mathbb{P}_d,$$

where $\mathbb{P}$ is the space of all real polynomials and $\mathbb{P}_d$ is the space of all real polynomials of degree less than or equal to d. Moreover, we can say that a quadrature rule has a precise degree of exactness d if it has a degree of exactness of d, but not of $d+1$, i.e., $\exists p \in \mathbb{P}_{d+1}; R_n(p) \neq 0$.

Definition 15 *Any quadrature rule [1] with degree of exactness $d = n - 1$ is of the interpolatory type. Moreover, a quadrature is an interpolatory one if and only if its function f can be obtained through interpolation, i.e.*

$$f(x) = \sum_{i=1}^{n} f(x_i)\, l_i(x), \tag{3.2}$$

which is the Lagrange interpolation formula, where

$$l_i(x) = \prod_{\substack{i=0 \\ i\neq j}}^{n} \frac{(x - x_i)}{(x_j - x_i)} = \frac{\rho(x)}{(x - x_i)\rho'(x_i)} \tag{3.3}$$

is the fundamental polynomial of the Lagrange formula. It logically follows that $\rho(x) = \prod_{i=0}^{n}(x-x_i)$.

Definition 15 is valid for any function $f = p \in \mathbb{P}_{n-1}$, that is, for any function equal to a polynomial of degree up to $n-1$, a set of n samples of this function is sufficient to uniquely determine this function. This is actually an important theorem in the interpolation theory field, and it is better detailed by Trefethen [24].

We are interested in knowing what are the necessary conditions for the quadrature rule to have a desired degree of exactness. That is, when can we say that the integral of a given function $f(\cdot)$ in an arbitrary measure $d\lambda$ is exactly equal to a weighted sum of samples of this function? This interest comes from the fact that given the special case of $f(\cdot)$ as a monic polynomial, one can clearly notice the resemblance of this problem with the one of the UT calculation, from Equation 2.14, by comparing the following two equations:

$$\int_{-\infty}^{\infty} x^k p_{\boldsymbol{x}}(x)dx = \sum_{i=0}^{n} s_i^k w_i \tag{3.4a}$$

$$\int_{a}^{b} f(x)d\lambda(x) = \sum_{i=0}^{n} \lambda_i f(x_i) \tag{3.4b}$$

For f equals to a monic polynomial (i.e., $f(x) = x^k,\ for\, k \in \mathbb{Z}^+$), and the measure $d\lambda$ equals to the probability distribution $p_{\boldsymbol{x}}(x)dx$ of the signal, both equations 3.4a and 3.4b mathematically denote the same process and both solutions will determine the same set of nodes $x_i = s_i$ sigma-point and weights which are going to be used for determining the quantizer's output levels and thresholds (th_n) respectively[2].

[1] in the sense of Definition 13

[2] Note that the quadrature denoted in Equations 3.4 is an n+1-point quadrature rule, according to definition 13.

3.1.1 Example

To illustrate this comparison between UT and Interpolatory Quadratures, let us consider a simple example case in which the measure of the quadrature integral $d\lambda(x)$ is equal to dx. This implies that the signal's PDF $p_{\boldsymbol{x}}(x)$ denotes an uniform distribution of probabilities within an interval $[a, b]$, where it is defined. Moreover, since we are free to choose whichever nodes and weights that satisfies both Equations 3.4a and 3.4b, let us choose the nodes $x_i = s_i$ equally spaced in the interval of the partition $S_n : a \leq x_1 < x_2 < ... < x_n \leq b$.

Within this determined conditions, we just stated the problem of the Newton-Cotes quadrature rule, where the resulting weights λ_i are called Cotes numbers (see [21] for more details on this computation). However, as discussed in Trefethen's works (both [24] and [25]), this is a very non-efficient method for choosing the quadrature nodes. As proved in his works and even before[3], the convergence of this finite summation when the nodes are equispaced diverges exponentially as n grows. Aside from the instability issue, this method is also a bad choice because it is a too signal-specific method, for it only accepts signals with a uniform distribution of probability.

With that in mind, we will propose in the following sections some more sophisticated methodologies for solving these problems.

3.2 Orthogonal Polynomials

To initiate this section it is important to state some new definitions which will be used throughout the rest of this thesis as an important methodology for calculating quadratures (and hence UTs as well). We will begin by defining orthogonality of polynomials, based on their inner products (as used in [21] and [22]).

Definition 16 *Let $\lambda(x) \geq 0, \forall x \in \mathbb{R}$, and $|\lim_{x\to\infty} \lambda(x)| < \infty$ and $|\lim_{x\to-\infty} \lambda(x)| < \infty$. Also, assume that the induced positive measure $d\lambda$ has finite moments of all orders. Then, for any two polynomials $u, v \in \mathbb{P}$, their inner product with respect to the measure $d\lambda$ is defined as:*

$$< u, v >= \int_{\mathbb{R}} u(x)v(x)d\lambda(x), \tag{3.5}$$

such that when $< u, v >= 0$, u is said to be orthogonal to v, for $u \neq v$. If $u = v$, then $< u, u >= ||u||^2$, which is the squared norm of u ($||u|| \geq 0, \forall u \in \mathbb{P}$).

As discussed by Szego [22], Gautschi [21] and many other works in the field of numerical analysis, we can use Definition 16 to define a family of monic orthogonal polynomials, such that

$$< \rho_i, \rho_j >= 0, for\, i \neq j;\; i, j \in \mathbb{Z}^+ \, and$$

$$||\rho_i|| > 0, for\, i \in \mathbb{Z}^+,$$

where every polynomial ρ_i is a monic one and can be determined by a *three-term recurrence relation.*

[3] It is a well known result since Carl Runge demonstrated its instability in the beginning of the 20^{th} century

Theorem 1 *Let $\rho_i, i = 0, 1, 2, ...$ be a family of monic orthogonal polynomials with respect to the measure $d\lambda$, then the three-term recurrence relation that defines them is;*

$$\rho_{i+1}(x) = (x - A_i)\rho_i(x) - B_i\rho_{i-1}(x) \tag{3.6a}$$

$$\rho_{-1}(x) = 0, \ \textit{and} \ \rho_0(x) = 1,$$

where

$$A_i = \frac{< x\rho_i(x), \rho_i(x) >}{||\rho_i(x)||} \tag{3.6b}$$

$$B_i = \begin{cases} 1 & \textit{for } i = 0 \\ \frac{||\rho_i(x)||^2}{||\rho_{i-1}(x)||^2} & \textit{for } i>0 \end{cases} \tag{3.6c}$$

The proof of this theorem is stated in [21], at the section presenting the recurrence relation itself.

Another important theorem is presented by Szego [22] (in the section presenting the Gauss-Jacobi quadrature problem), which states that

Theorem 2 *If $x_1 < x_2 < ... < x_n$ denote the zeros of an orthogonal polynomial $p_n(x) \in \mathbb{P}_n$, there exists a set of real numbers $\lambda_i, i \in [1, n]$, such that:*

$$\int_a^b f(x)d\lambda(x) = \sum_{i=1}^{n} f(x_i)\lambda_i, \tag{3.7}$$

for any given $f \in \mathbb{P}_{2n-1}$. The measure $d\lambda(x)$ and the integer n uniquely defines the set of real numbers λ_i.

This theorem implies that whenever we have a set of points x_i which denote the zeros of a polynomial $p_n(x)$, they can be used as nodes of a quadrature rule determined by a given measure $d\lambda(x)$. More than that, we can have the function whose integral being is being approximated of an order up to $2n - 1$ for a set of n points. This means that if we apply this theorem in a context of moment preserving UT calculation, we can conserve moments of the order up to $\alpha = 2n - 1$ with just n sigma-points. This case is known as the Gauss quadrature rules [21], which is a method of quadrature implementation which maximizes the order of integration.

Those last two theorems are arguably the most important mathematical support for Gaussian quadratures implementation, in general, and are the main support for the works of Medeiros [7], in which this thesis is greatly based upon. They also break down the problem of UT calculation to solving Quadrature problems finding nodes and weights. The following table presents some important orthogonal polynomials and their respective three-term recurrence relations used for quadrature calculations.

[†] This is the formulation for Gauss-Chebyshev polynomials of the First Kind, refer to [21] or [22] for more details on the Second, Third and Fourth Kinds, which are of less importance in the context of this work.

Table 3.1: Recurrence relations of classical orthogonal polynomials for a given measure $d\lambda(x)$

.

Name	$d\lambda(x)$	A_k	B_0	B_k, $(k \geq 1)$
Jacobi	$(1-x)^{\alpha}(1+x)^{\beta}dx$	A_k^J	B_0^J	B_k^J
Legendre	$1dx$	0	2	$\frac{1}{4-k^{-2}}$
Chebyshev†	$(1-x^2)^{-\frac{1}{2}}dx$	0	π	$\frac{1}{2}(k=1), \frac{1}{4}(k>1)$

$$A_k^J = \frac{\beta^2-\alpha^2}{(2k+\alpha+\beta)(2k+\alpha+\beta+2)}^{*}$$

$$B_0^J = \frac{2^{\alpha+\beta+1}\Gamma(\alpha+1)\Gamma(\beta+1)}{\Gamma(\alpha+\beta+1)},\ B_k^J = \frac{4k(k+\alpha)(k+\beta)(k+\alpha+\beta)}{(2k+\alpha+\beta)^2(2k+\alpha+\beta+1)(2k+\alpha+\beta-1)}^{**}$$

It is important to state that all of the quadrature rules resulting from those measure functions stated in Table 3.1 are defined in the interval $[-1,1]$, but can be extended for any interval $[a,b]$ with just some normalization of the integrands (see [21]).

Moreover, one can observe that the Legendre Polynomials define a quadrature rule for the same measure as the Newton-Cotes case exemplified in the previous section. However, since the roots of Legendre polynomials are not equispaced, they do not suffer from the same instability problems as the Newton-Cotes, therefore presenting more accurate and efficient results for large n.

One last important comment about Table 3.1 refers to the relation between the Jacobi polynomials and the Legendre and Chebyshev ones. An observant reader can note that those last two classes of polynomials are just special cases of the Jacobi formulation: Legendre is the case where $\alpha = \beta = 0 \Leftrightarrow d\lambda(x) = 1dx$; and Chebyshev polynomials of the First kind is when $\alpha = \beta = -0.5 \Leftrightarrow d\lambda(x) = (1-x^2)^{-\frac{1}{2}}dx$.

We are specially interested in the Chebyshev polynomials, because of their importance in approximation theory and the efficient numerical methods existent for calculating quadratures of this genre, first noticed by Clenshaw and Curtis in their 1960 work [26], and latter on extensively used as basis for researches in better algorithm implementations (such as in [27] and [25]). We will now refer to the method presented by Clenshaw and Curtis as the Clenshaw-Curtis Quadrature rule (CC Quadrature), and present in the next section some comparisons between this method and the Gauss-type quadratures, highly based on the analysis present by Trefethen in [23].

[*] If $k = 0$, then the common factor $\alpha + \beta$ in the numerator and denominator of α_0^J should be cancelled (actually must be if $\alpha + \beta = 0$)

[**] If $k = 1$, then the last factors in the numerator and denominator of β_1^J should be cancelled (actually must be if $\alpha + \beta + 1 = 0$)

3.3 Gauss Quadrature vs. Clenshaw-Curtis

Both methods (Gauss and CC) come as a better alternative than Newton-Cotes formula for quadrature calculation, since they converge for any continuous integrand f, and do not suffer from Runge phenomena. Gauss-type quadratures present the most efficient method regarding the order of integration for a given set of n points. They also overcome the Newton-Cotes method (which has a algorithm complexity of $O(2^n)^\dagger$) by presenting an algorithm (proposed by Golub and Welsch [28]) with complexity of $O(n^2)$, acquired by calculating the eigenvalues and eigenvectors of a tridiagonal matrix, which are related to the nodes and weights (respectively) of the quadrature.

The CC Quadrature, on the other hand, presents a family of formulas based on sampling the integrand f at the Chebyshev points (which will be defined on the next chapter) and calculating the Chebyshev coefficients via FFT (the Fast Fourier Transform), which makes this method implementable with a complexity of $O(n \cdot log\, n)$. However, like the Newton-Cotes method, the CC quadrature integrates polynomials exactly of order up to $n-1$ for a given set of n-points quadrature.

For that, it seems that the CC quadrature is faster than and as robust as the Gauss, but "half as efficient". However as pointed in the works of O'Hara and Smith [29] in 1968 (and latter on analyzed by Trefethen in [23]), for many the integrands both methods turn out be equally accurate. Since the number of points n of the quadrature are actually related to the number of quantization levels of the proposed design, doubling this set of points could be done by just adding one more bit to the quantizer (this argument will be more deeply explained in the following chapter). Therefore, we will make use of the faster implementation of the CC quadrature which uses the FFT algorithm for a faster and as efficient method of acquiring the nodes and weights necessary for the quantizer design.

Finally, it is also interesting to shine a light on the fact that the Chebyshev points presents a distribution more concentrated on the extremes of its range. As we can observe in the following illustration. This behaviour is going to be better analyzed in the next chapter as we detail the behaviour of the proposed design in a case study scenario.

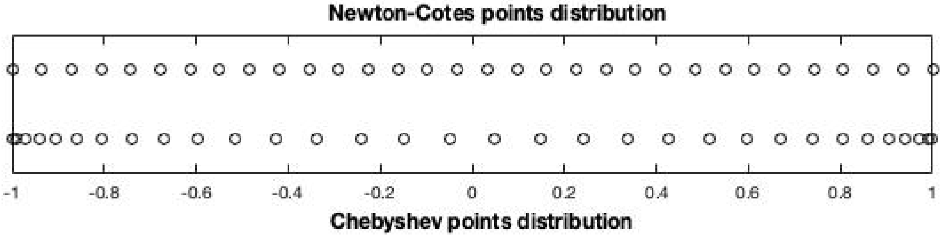


Figure 3.1: Comparisons between distributions of nodes for different quadrature methods. On the top, the 32 points of a Newton-Cotes distribution of points for the interval $[-1, 1]$, and on the bottom the distribution of Chebyshev points of 1^{st} Kind in the same interval.

[†] Big O notation refers to order of complexity of algorithms, see [30] and [31] for more details.

Chapter 4

Case Study

This chapter is the core of this work, where we present a case study and simulations regarding the quantizer design mentioned in the Introduction. We will first detail the methodology used for calculating the parameters of the quantizer, based on the theories presented by the previous chapters, contextualizing the quantization process with moment preserving UT calculation by means of Numerical Interpolatory Quadrature computation.

Then, we present the main contribution of this thesis, which is the Modified Direct Method for sigma-points and weights calculation of the UT, utilizing the CC Quadrature technique. We follow by better defining the FOM proposed for analyzing nonlinear moment preserving quantization processes, mentioned at the end of Chapter 1. And finally, we present some simulation results regarding a case study scenario of a tonal sine wave as input signal.

4.1 Quantizer Design

Designing an N-level quantizer can be broken down into defining its N *output levels* and N-1 *input thresholds*. For that, we will use as a basis the design methodology proposed by Delp in his 1990 paper [32], which was later on used also by Medeiros on his thesis [7]. This method constructs a quantizer in which the *output levels* are related to UT's sigma-points (acquired from the nodes of a quadrature calculation), and the *input thresholds* are related to UT's weights.

Differently from those previous works, however, we will use the CC Quadrature to define *a priori* the nodes of the quadrature as the Chebyshev Points (Cheb-points), which will consequently predefine the quantizer's *output levels*. This choice was made primarily to make the DAC process independent of the input signal, thus making at least the Digital-to-Analog part of the converter signal-generic. This was not the case for the quantizer designed by Medeiros [7], which had to calculate specific output levels for each signal input. The specific choice of the Cheb-points was thought in order to avoid the Runge phenomena present in equally spaced points, and finally to match the CC quadrature used in the Modified Direct Method to calculate the signal moments.

The input threshold points th_j (with $j = 1, 2, ..., n-1$), however, are constructed following the same idea proposed by the mentioned authors. The accumulated probability of the input signal in between the intervals of th_{j-1} and th_j is numerically equal to the weight associated with the sigma-point of index j. Figure 4.1 illustrates very well this constructive process by presenting the characteristic curve of a 4-level quantizer in which its input signal is a generic sine wave, whose PDF is equal to an arcsine distribution[1]denoted by $p_{\boldsymbol{x}}(x)$.

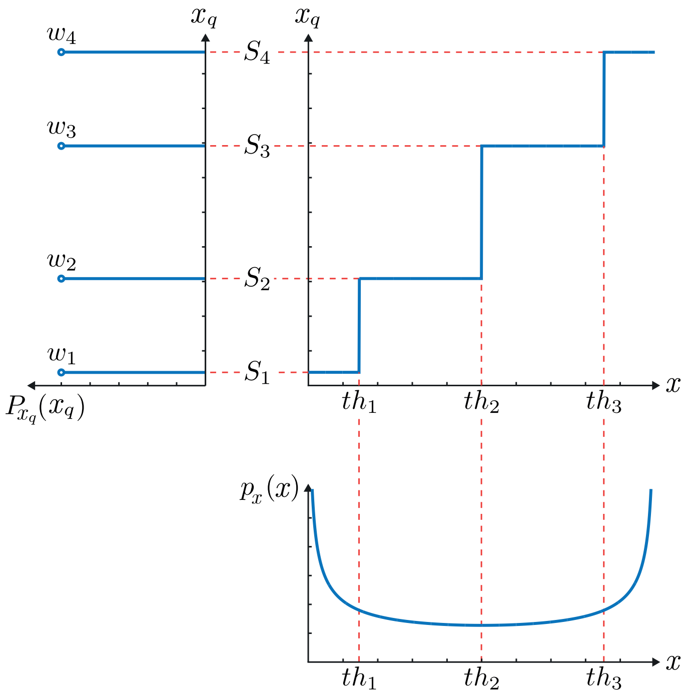


Figure 4.1: Characteristic curve of a 4-level quantizer, denoting an arcsine probability distribution $p_{\boldsymbol{x}}(x)$ as input, and its discrete counterpart $p_{\boldsymbol{x_q}}(x_q)$ resulted from the UT calculation of $p_{\boldsymbol{x}}$.
Source: Image taken from [7], with author's consent

We can now mathematically write the construction statements made on the previous paragraph as a probability equation:

$$\begin{aligned} & P(th_{j-1} < x < th_j) = \int_{th_{j-1}}^{th_j} p_{\boldsymbol{x}}(x)dx = w_j \\ \Leftrightarrow \quad & \int_{-\infty}^{th_j} p_{\boldsymbol{x}}(x)dx - \int_{-\infty}^{th_{j-1}} p_{\boldsymbol{x}}(x)dx = w_j \\ \Leftrightarrow \quad & F_{\boldsymbol{x}}(th_j) - F_{\boldsymbol{x}}(th_{j-1}) = w_j \end{aligned} \tag{4.1}$$

where $P(th_{j-1} < x < th_j)$ is the probability of the signal input x to assume a value in between the values th_{j-1} and th_j, and $F_{\boldsymbol{x}}(th_j) = P(x < th_j)$ is the signal's CDF.

Moreover, we can state that $F_{\boldsymbol{x}}(th_1) = \int_{-\infty}^{th_1} p_{\boldsymbol{x}}(x)dx = w_1$, and by using Equation 4.1 to show

[1] This is a commonly known result in Probability Theory, see [11] or [12] for more details on this modeling process.

the particular case of $j = 2$, we have:

$$\begin{aligned} F_{\boldsymbol{x}}(th_1) = w_1 &\Leftrightarrow th_1 = F_{\boldsymbol{x}}^{-1}(w_1) \\ F_{\boldsymbol{x}}(th_2) - \underbrace{F_{\boldsymbol{x}}(th_1)}_{w_1} = w_2 &\Leftrightarrow th_2 = F_{\boldsymbol{x}}^{-1}(w_2 + w_1) \end{aligned}$$

Therefore, by induction, we arrive at the threshold expression in terms of the probability weights:

$$th_j = F_{\boldsymbol{x}}^{-1}\Bigg(\sum_{i=1}^{j} w_i\Bigg), \tag{4.2}$$

where $F_{\boldsymbol{x}}^{-1}(F_{\boldsymbol{x}}(x)) = x$ is known as the quantile function (see [11]), and it is assumed to exist for every input signal we are going to work with.

4.2 Modified Direct Method

In this section, we present an alternative to the method priorly detailed in Section 2.2.1 of this work. More specifically, we are now trying to avoid the problem presented by Equation 2.17, which was proven to be extremely ill-conditioned. The proposed solution came as a consequence of pre-defining the sigma-points s_i as the Cheb-points, because that matrix appeared solely to calculate the UT sigma-points.

We now have the problem of calculating the weights associated to those sigma-points in order to satisfy Equation 2.14. For that, let us consider, without loss of generality, the formulation initially proposed by the Classic Direct Method in which we analyzed a set of 3 pairs of s_i, w_i.

$$\begin{aligned} s_1^0 w_1 + s_2^0 w_2 + s_3^0 w_3 &= m_0 \\ s_1^1 w_1 + s_2^1 w_2 + s_3^1 w_3 &= m_1 \\ s_1^2 w_1 + s_2^2 w_2 + s_3^2 w_3 &= m_2 \end{aligned}$$

Note, however, that now we only need to have knowledge of the first 3 moments[2], since we only have 3 variables (the weights). Writing the previous equations in matrix form leads to:

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 \\ s_1 & s_2 & s_3 \\ s_1^2 & s_2^2 & s_3^2 \end{bmatrix}}_{\mathbf{S}_{3x3}} \underbrace{\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}}_{\mathbf{W}_{3x1}} = \underbrace{\begin{bmatrix} 1 \\ m_1 \\ m_2 \end{bmatrix}}_{\mathbf{M}_{3x1}} \tag{4.3}$$

where $\mathbf{S}_n$ is the Sigma matrix for an n-set of points, $\mathbf{W}$ and $\mathbf{M}$ the Weights and Moments matrices, respectively.

[2] Actually, we only need to calculate the first and second order moments, since the 0-th order moment is, by definition, equal to one.

We now have a signal-specific solution that only involves the knowledge of the input's statistical moments, and a system of linear equations in order to calculate its associated UT. The moments can be calculated numerically with only the signal's samples as information, but, in the context of this case study (in which we are analyzing a sine wave input), we can derive an analytic form to calculate this moments by definition using CC quadratures.

$$m_k = \int_a^b f(x) p_{\boldsymbol{x}}(x) dx, \tag{4.4}$$

where $f(x) = x^k$ and $p_{\boldsymbol{x}}(x) = \frac{1}{\pi\sqrt{1-x^2}}$, which is the PDF of a tonal sine wave (of an arbitrary frequency) with unit amplitude ($g(x) = sin(2\pi f_0 x)$) defined in a generic interval $[a, b]$, that we will assume to be from -1 to 1.

Equation 4.4 is by definition the calculation of the k-th moment of a sine wave input signal, and it is also a quadrature problem with measure $d\lambda(x) = p_{\boldsymbol{x}}(x)dx$, which is exactly the definition we presented in Table 3.1 of the Gauss-Chebyshev quadrature. We will solve this quadrature by means of the CC method, which was computationally implemented and made available by Trefethen as a MATLAB open source library code named *chebfun* (see [33] and [25] for more details on this library). This formulation, however, presented numerical issues during its implementation for Sigma matrices of order bigger than 5x5, i.e., for a quantizer design of more then 32 levels (which would have resolution of 5 bits only). We will better detail this problems in Section 4.4.

4.3 Moment Preserving Ratio

In this section, we present the idea of a FOM capable of measuring how much the quantizer fulfilled its purpose. That is, since we are designing a quantizer which is intended, by construction, to preserve the statistical moments of the input signal, why don't we measure the output's statistical moments and compare to the ones we calculated for the input signal?

We hereby propose a simple implementation for a Moment Preserving Ratio (MPR), such that

$$MPR_{k\%} = 100 \cdot \sqrt{\left(\frac{m_{input_k} - m_{output_k}}{m_{input_k}}\right)^2}\%, \tag{4.5}$$

where m_{input_k} and m_{output_k} are respectively the k-th moment of the input signal and output signal. Figure 4.2, represents the result of a series of MPR values calculated for a linear quantizer and for a CC quantizer, whose input signals were both a tonal unit amplitude sine wave. The graphic was ploted in log scale to evidentiate the minimum amount of error the CC quantizer presented in this metric for up to the 100^{th} moment, which in linear scale would appear just as zeros.

Obviously, this FOM is not appropriated for a linear quantizer, therefore the high relative errors compared to the results of the CC quantizer, which was already expected since linear quantizers in general completely alter the statistical structure of the input signal [34]. Also, this plot can represent a scale of comparison, where we observed in simulation that the quantizer in fact conserved the

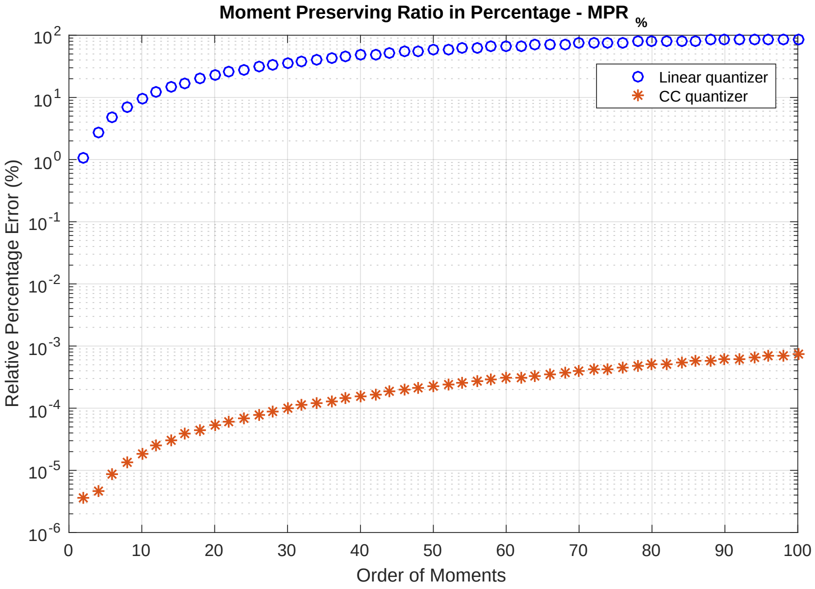


Figure 4.2: MPR values for a 5-Bit Quantizer. The circles represent the values of the MPR$_{\%}fora5-Bitlinearquantizer,andasterisksrepresenttheMPR_{\%}for5-BitCCquantizerdesigned$

moments up to the 31^{st} order (which was the expected theoretical result for a 5-Bit quantizer) and even more. As can be seen in Figure 4.3, the moments started to show any significant divergence between the input and output moments only after the 400^{th} moment, and significant divergences (more than 10% of MPR) only after the 700^{th} moment. That again proved the Modified Direct Method was able to construct a quantizer model that satisfies Equation 2.14 with an even higher level of robustness than expected (at least for the sinusoidal input simulated case).

4.4 Simulation Results

Here we present some simulations results regarding the other metrics stated in Chapter 1 (SNR, SINAD and SFDR) and the behaviour of quantizers with different resolutions until failure resolution is achieved.

First, we present Figures 4.4(a) and 4.4(b), which demonstrate a first general comparison between the linear quantizer and the designed one (respectively). The designed quantizer's characteristic curve shows a disposition of output levels that is concentrated on the extremes of the range interval, as was expected both by the natural behaviour of points distribution from the Chebyshev points function, and by the already presented result in the works of Medeiros [7]. This disposition of points can be of use in general nonlinear application of data conversion processes, specially if we take into consideration that the linear quantizer itself normally does not represent well the conversions on its extreme points and our designed quantizer does present a better refinement in these areas.

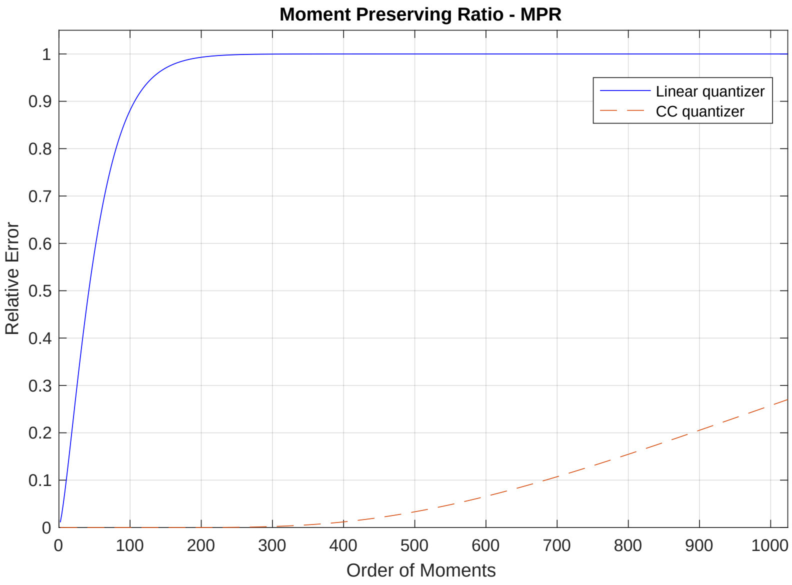


Figure 4.3: Linear Scale representation of the MPR values for a 5-Bit Linear quantizer (continuous curve) and for the CC quantizer designed (dashed curve).

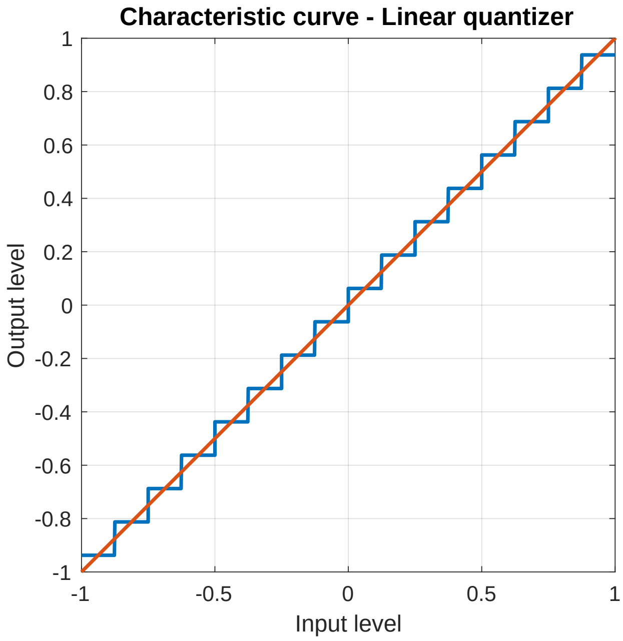
Although the disposition of output levels are not equally spaced, one can prove that for this special case of a sinusoidal input, the intervals between each threshold point is of the same length[3]. Which does not present any advantages or disadvantages *per se* , but it is an interesting behaviour.

Next, we present some results based on classical dynamic metrics to reafirm the results observed by Medeiros, which is serving in the current section of this work as a benchmark for the proposed method results. Figures 4.5(a)-4.5(b) and Figure 4.6 show results identical to the theoretically expected for the simulated range of resolutions, based on the cited work and considering that we are analyzing the spectral interval of the signal until the 10^{th} harmonic.

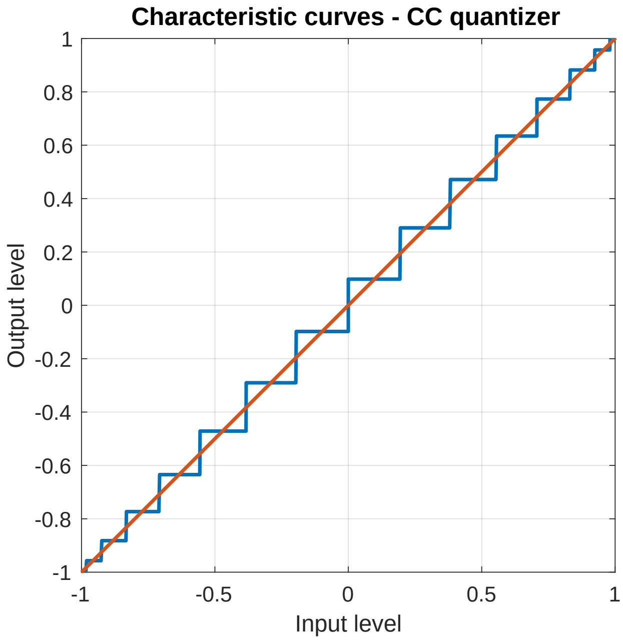
In Figures 4.5(a) and 4.5(b), the curves in blue with triangle points represents the behaviour resulted of the linear quantizer conversion process. The yellow line with square points represents the theoretical calculation of the SNR based on a AWGN noise distribution over the frequency interval of the signal plus the quantization distortions, calculated with the formula $SINAD = 6.02 \cdot N + 1.76\, dB$ (where N is the resolution of the ADC). And the orange line with circular points is the measured SNR of the output signal after the CC quantizer application. We can notice that the three curves are very little apart from each other (2 dB apart at most) in both metrics, which leads to the conclusion that at least for this range of resolution the application of the proposed nonlinear quantizer does not present any improvements.

The most expressive improvement is presented by the SFDR curve, Figure 4.6, where we can only start to observe some advantages for more than 3-Bit resolution. Naturally, before that, any converter design would present too much error to even consider being used in practical terms. For 4

[3] see [7], section 4.3 for prove of this result



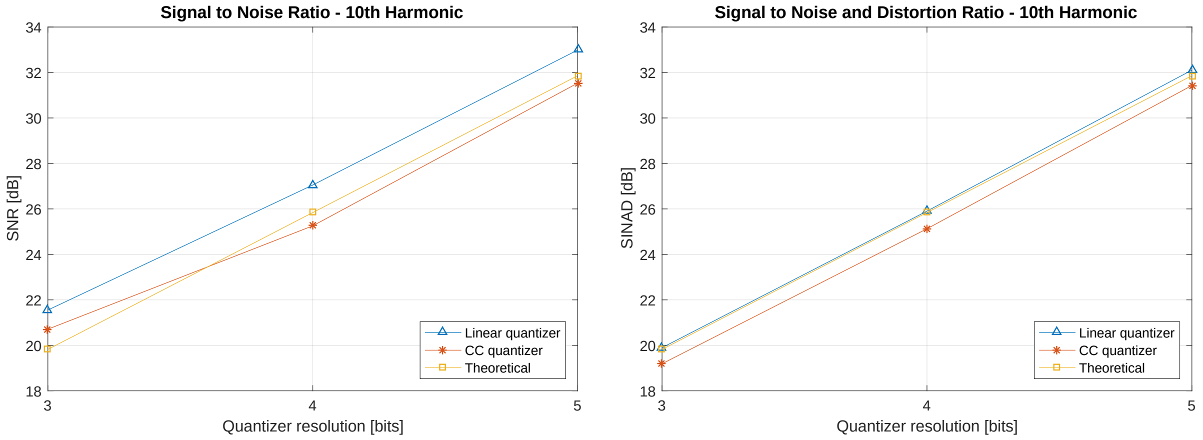
(a) Linear Quantizer



(b) CC Quantizer

Figure 4.4: Characteristic curves of a) Linear Quantizer and b) CC Quantizer, both with 4-Bit resolution for better visualization of the disposition of thresholds and output levels

and 5-bit resolution, however, we can observe an improvement of around 15 dB more range without any distortion, which is a promising result, indicating we might be able to implement design with even higher ranges in better resolutions.



(a) SNR

(b) SINAD

Figure 4.5: a) Signal-to-Noise Ratio (SNR) and b) Signal-to-Noise and Distortion Ratio (SINAD) presented for 3, 4 and 5 bit resolution quantizer analyzed up to the 10^{th} harmonic.

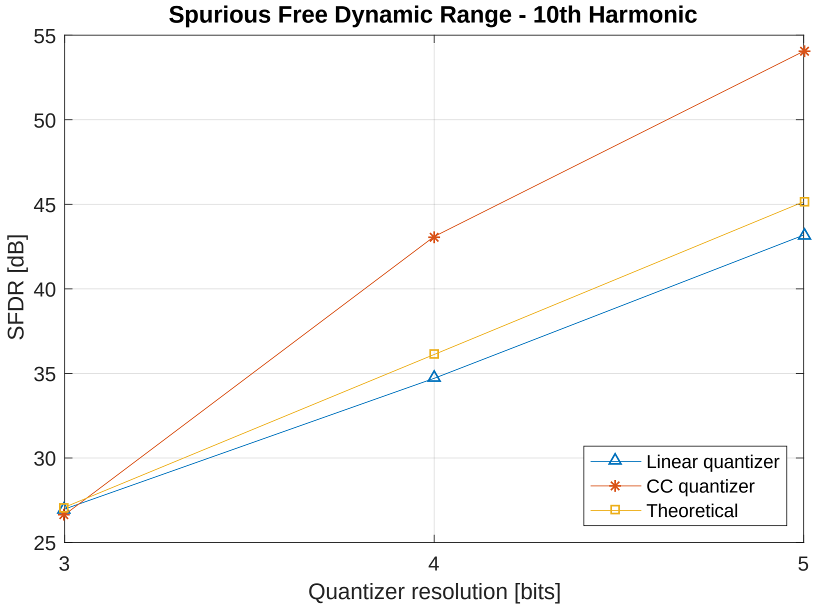
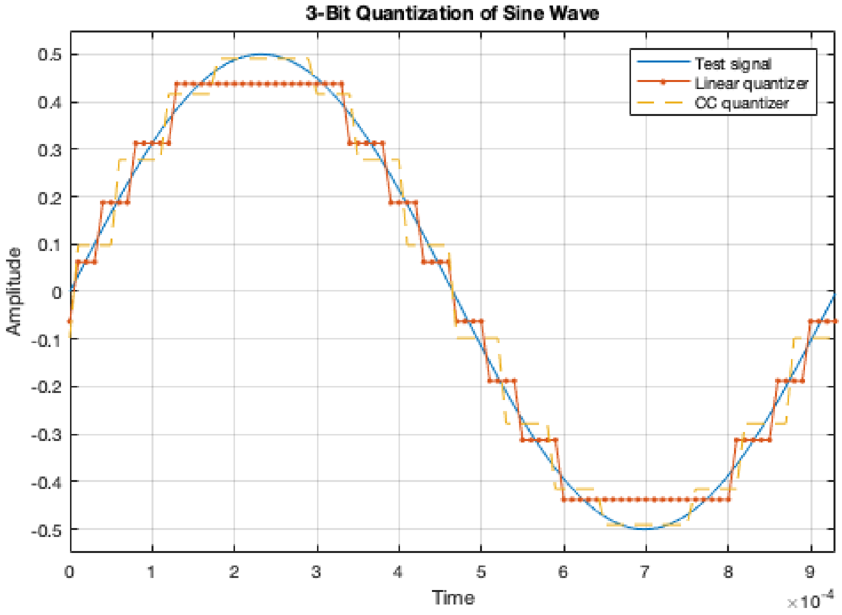


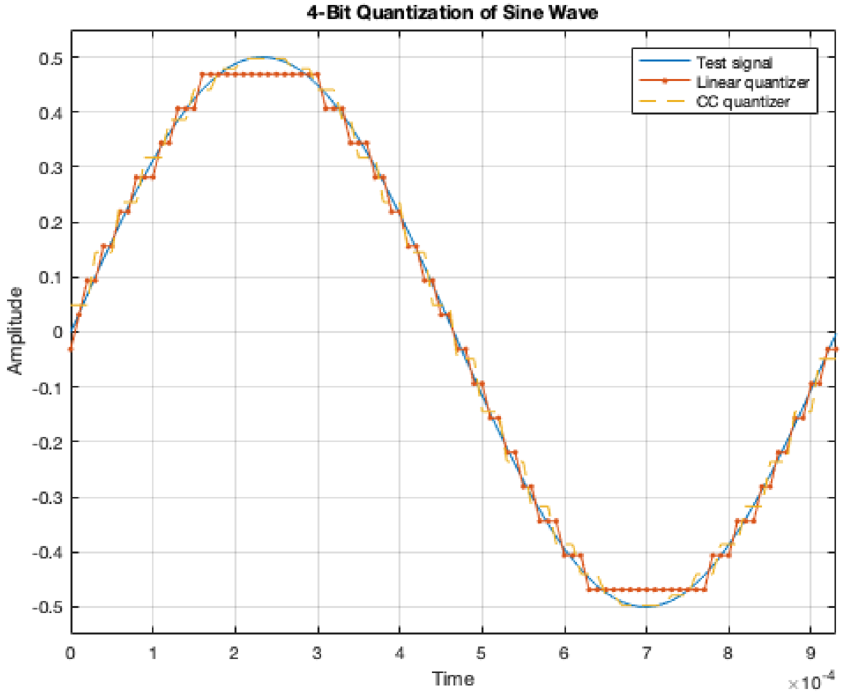
Figure 4.6: Spurious-free Dynamic Range presented for 3, 4 and 5 bit resolution quantizer analyzed up to the 10^{th} harmonic.

The combined interpretation of Figures 4.5(a),4.5(b) and 4.6 also showed an important behaviour: the quantizer did not eliminate the distortions, it just distributed them in a different manner along the signal spectre. In fact, from the results of Medeiros work [7], we can observe that this quantizer "pushed" the spurious harmonics to the higher frequencies.

The following set of figures demonstrate the transient curves of both the linear quantizer and the designed CC quantizer in superposition to the test signal for a more intuitively analysis of the quantization process. Until we start observing the resolution issues mentioned, it is notorious that the curve in yellow (representing the CC quantizer) clearly approximates better the test signal than the curve in orange (which represents the linear quantizer) for the same resolution. However, we started observing some singularities errors during the computing of the Weights matrix.

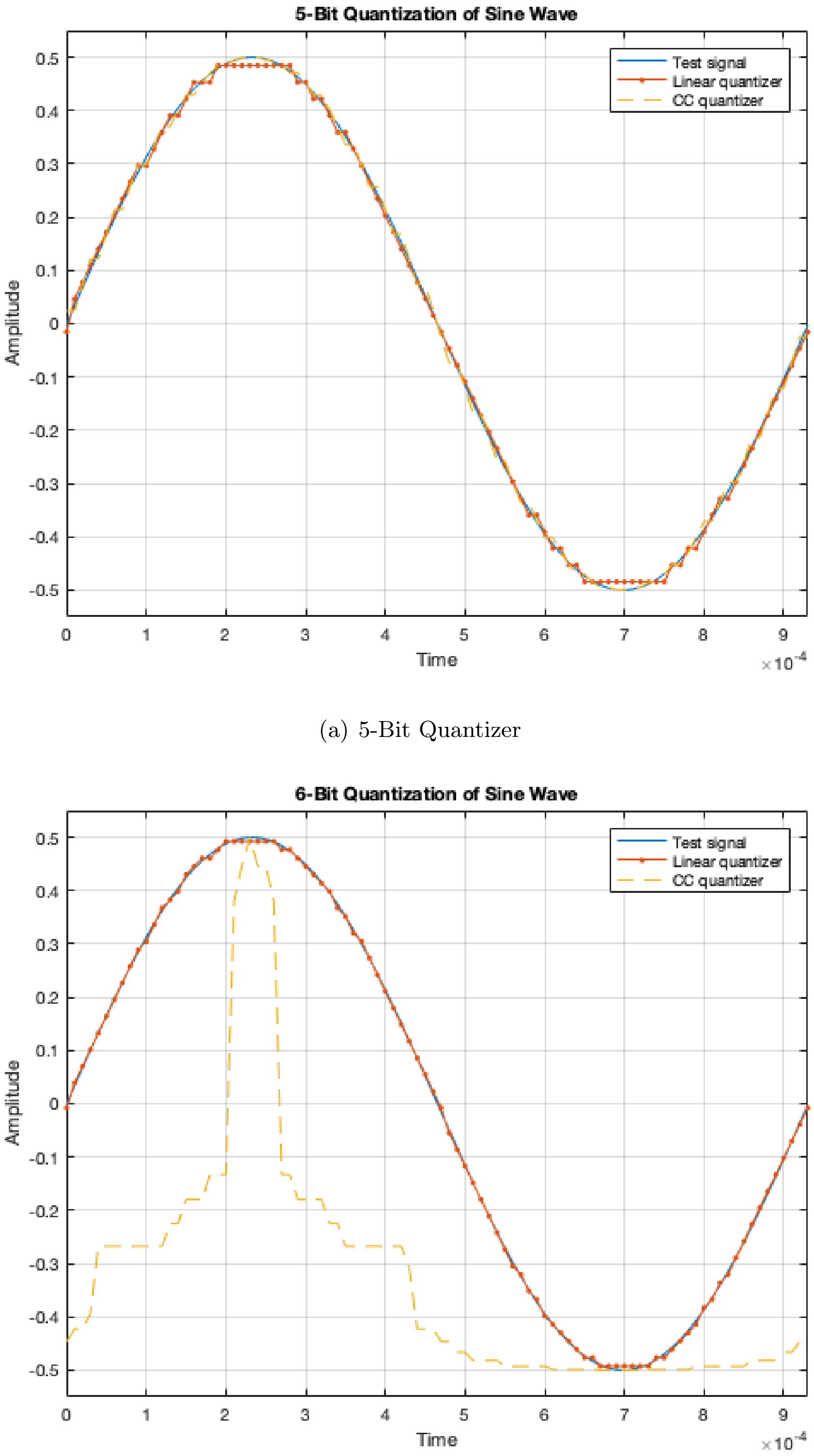


(a) 3-Bit Quantizer



(b) 4-Bit Quantizer

Figure 4.7: 3 and 4-Bit transient quantizer's curves. In yellow the CC quantizer curves tracking the test input signal (blue), and in orange the linear quantizer behaviour.



(a) 5-Bit Quantizer

(b) 6-Bit Quantizer

Figure 4.8: a) 5-Bit quantizers, representing the highest resolution attained without errors with the proposed Modified Method; and b) the 6-Bit linear quantizer tracking the test signal input, and the CC quantizer failing on doing so after loosing reference during weight calculation.

Theoretically, we could just invert the Sigma matrix and solve the linear problem, however inverting matrices is a historically difficult process to compute numerically. The approach, then, involved some series of matrices diagonalizations applied adaptatively by the operator $W = S\backslash M$, in MATLAB. Even that could get the simulation no far than the 6-Bit resolution shown. The problem appears to be the fact that, in the same linear system, we have matrices elements of an increasing order of magnitude in **S**, whereas the elements in **W** are generally small (remember that the sum of all weights is 1) and the Moments matrix itself can be very sparse depending on the distribution of probability of the signal analyzed. A symmetrical signal, for instance, has all its odd moments equal to 0, which makes the matrix of moments have at least all of its elements equal to 0.

All these factors leads us to the conclusions presented in the next chapter, while still presenting alternative ways of getting around these problems with suggested approaches.

Chapter 5

Concluding remarks

The quantization is an extremely important process for any kind of computer to operate, since the Digital approach actually gives computers information in a context in which they can actually work on (bits, 0's and 1's). Also, in the Digital domain we have the possibility to carry complex calculations that are not obvious on the Analog domain, plus we have the easiness to adapt the algorithms to changing circumstances [35]. For that, nowadays analog devices are rarely the best option, and almost everything that involves technology, from TV's to rockets, has this kind of conversion process involved.

This thesis provided basic concepts involving ADCs in Chapter 1, such as sampling, quantization and coding definitions and theories, and metrics for ADC's performance evaluation. Afterwards, in Chapter 2, we detailed some crucial concepts of Probability Theory, mainly involving probability distributions and central statistical moments classical definition. That was made in order to formulate the definition of the Unscented Transform used throughout this work. At the end of this chapter a method for computing UT calculation of a generic input signal was shown as an example of possible calculation method, but that had some fundamental flaws.

In Chapter 3, we presented the concepts of Numerical Interpolatory Quadratures and how that with orthogonal polynomials theory could be an important mathematical tool used for UT calculation. Then, in Chapter 4, we presented the most important part of this thesis, which is the actual Design of a nonlinear quantizer based on numerical quadratures, using Delp's [32] idea as basis.

The design followed a slightly different approach than Delp's [32] and Medeiros' [7] propositions, but still sharing the main idea. We proposed the Modified Direct Method for that, which had some promising theoretical advantages for practical implementations, such as being a signal generic method, and presenting an *a priori* computationally easy solution. However, as shown in the Case Study chapter, the simulation results revealed that Equation 4.3 still presented some numerical issues for implementations of order of resolution bigger than 5 bits.

However, within the range of resolution in which this numerical problems were not significant, the Modified Direct Method presented results that led to a nonlinear quantizer design whose performance surpasses that of a common linear quantizer implementation. The proposed metric (the Moment Preserving Ratio - MPR) was also used to gauge both linear and nonlinear quantizers behaviour, and proved to work well to evaluate this class of Moment Preserving Quantizers.

5.1 Future Work

In the end, the proposed method cannot be implemented in the current proposed form, because of the instability issues regarding the singularities present in quantizers of 6 bits or more. To overcome that, some new ideas can be proposed as future works.

The first, and most obvious, idea is to actually better investigate what specifically causes the numerical approximation issues in the solution of Equation 4.3. If the issue is actually originated from the fact that the system is working with numbers with very different orders of magnitudes, then a possible work around would be to research in a way of mapping these elements to a mathematical field in which their magnitudes are closer to each other.

Another possible idea is to invest in the method proposed by Medeiros [7], but instead of working with the Gauss-Chebyshev (or other Gauss specific) polynomials, we can try to apply the Gauss-Jacobi quadrature, whose polynomials presented in Table 3.1 are the most generic. With that, the real challenge is to find a mathematical expression in terms of α and β, that maps this generic polynomial representation to any specific input signal distribution.

And finally, the last idea proposed is an actual algorithmic update in the implementation done by Medeiros [7]. In his thesis, it was mentioned that the order of the used algorithm complexity was exponential, mainly due to the gaussian quadrature implementation based on the eigenvalues and eigenvectors of the Jacobi tridiagonal matrix. This method was proposed by Golub and Welsch [28] in 1967. However, there are way more efficient and faster methods for gaussian quadrature implementation, such as the one proposed by Glaser, Liu and Rokhlin in 2007 [36] (which appeared as a solution in a context of differential equations). Another one is the algorithms proposed by Hale and Townsend in 2013 [37], or the papers of Bogaert (such as [38]) which implements an approximately $O(n)$ algorithm using asymptotic formulas. These papers could be better studied an analyzed to bring a similar solution to the context of nodes and weights calculations for UT calculations.

Bibliography

[1] NYQUIST, B. Y. H. Certain Topics in Telegraph Transmission Theory. 1928.

[2] SHANNON, C. E. A Mathematical Theory of Communication. v. 5, n. I, p. 365–395, 1948.

[3] LATHI, B. P.; DING, Z. *Modern Digital and Analog Communication Systems.* [S.l.]: Oxford University Press, 2009. 995 p. ISBN 9780195331455.

[4] RHEW, H. G. et al. A fully self-contained logarithmic closed-loop deep brain stimulation SoC with wireless telemetry and wireless power management. *IEEE Journal of Solid-State Circuits*, v. 49, n. 10, p. 2213–2227, 2014. ISSN 00189200.

[5] SUNDARASARADULA, Y.; CONSTANDINOU, T. G.; THANACHAYANONT, A. A 6-bit, two-step, successive approximation logarithmic ADC for biomedical applications. *2016 IEEE International Conference on Electronics, Circuits and Systems, ICECS 2016*, p. 25–28, 2017.

[6] PAGIN, M.; ORTMANNS, M. Evaluation of logarithmic vs. linear ADCs for neural signal acquisition and reconstruction. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, p. 4387–4390, 2017. ISSN 1557170X.

[7] MEDEIROS, J. E.; HADDAD, S. A. Nonlinear quantizer design in data conversion systems using the unscented transform. *Proceedings - IEEE International Symposium on Circuits and Systems*, p. 0–3, 2017. ISSN 02714310.

[8] TEWKSBURY, S. et al. Terminology related to the performance of s/h, a/d, and d/a circuits. *Circuits and Systems, IEEE Transactions on*, v. 25, p. 419 – 426, 08 1978.

[9] RAZAVI, B. *Principles of Data Conversion System Design.* [s.n.], 1994. ISBN 9780470545638. Disponível em: <http://ieeexplore.ieee.org/xpl/bkabstractplus.jsp?bkn=5264233>.

[10] SANTOS, M.; HORTA, N.; GUILHERME, J. A survey on nonlinear analog-to-digital converters. *Integration, the VLSI Journal*, Elsevier, v. 47, n. 1, p. 12–22, 2014. ISSN 01679260. Disponível em: <http://dx.doi.org/10.1016/j.vlsi.2013.06.001>.

[11] PAPOULIS, A.; Unnikrishna Pillai, S. *Probability, Random Variables and Stochastic Processes.pdf.* Fourth. [S.l.]: McGraw-Hill Higher Education, 2002. 852 p.

[12] BILLINGSLEY, P. *Probability and Measure.* [S.l.: s.n.], 1986.

[13] JULIER, S.; UHLMANN, J. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, v. 92, p. 401 – 422, 04 2004.

[14] MENEGAZ, H. M. T. *Unscented Kalman Filtering on Euclidean and Riemannian Manifolds.* Tese (Doutorado) — Universidade de Brasília, 2016.

[15] JULIER, S. J.; UHLMANN, J. K. A consistent, debiased method for converting between polar and cartesian coordinate systems. *Proceedings of SPIE - The International Society for Optical Engineering*, 08 1997.

[16] Ortega P, A. E.; De Menezes, L. R.; ABDALLA, H. Statistical modeling of manufacturing uncertainties for microstrip filters. *Journal of Microwaves and Optoelectronics*, v. 10, n. 1, p. 179–202, 2011. ISSN 21791074.

[17] MENEZES, L. et al. Efficient computation of stochastic electromagnetic problems using unscented transforms. *Science, Measurement Technology, IET*, v. 2, p. 88–95, 04 2008.

[18] SHOHAT, J. A.; TAMARKIN, J. D. *The Problem of the Problem of Moments.* third. [S.l.: s.n.], 1943.

[19] TABATABAI, A. J.; MITCHELL, O. Edge location to subpixel values in digital imagery. *IEEE transactions on pattern analysis and machine intelligence*, v. 6, p. 188–201, 02 1984.

[20] JUNIOR, E. Alves da C. *Propagação de incertezas em eletromagnetismo.* Tese (Doutorado) — Universidade de Brasília, 01 2009.

[21] GAUTSCHI, W. Construction of gauss-christoffel quadrature formulas. *Mathematics of Computation - Math. Comput.*, v. 22, p. 251–251, 04 1968.

[22] SZEGÖ, G. *Orthogonal Polynomials.* [s.n.], 1997. 1–56 p. ISSN 09476539. ISBN 9781479923748. Disponível em: <papers2://publication/uuid/EBDE9D96-C7CD-40AC-B40E-DABBE94D2CFB>.

[23] TREFETHEN, L. N. Is Gauss Quadrature Better than Clenshaw–Curtis? *SIAM Review*, v. 50, n. 1, p. 67–87, 2008. ISSN 0036-1445. Disponível em: <http://epubs.siam.org/doi/10.1137/060659831>.

[24] TREFETHEN, L. N. *Approximation Theory and Approximation Practice.* [S.l.: s.n.], 2011. 163–171 p.

[25] TREFETHEN, L. N. *Spectral Methods in MATLAB.* [S.l.: s.n.], 2011. ISSN 0586-7614. ISBN 0898719593.

[26] CLENSHAW, C. W.; CURTIS, A. R. A method for numerical integration on an automatic computer. *Numer. Math.*, v. 2, p. 197–205, 12 1960.

[27] WALDVOGEL, J. Fast construction of the Fejér and Clenshaw-Curtis quadrature rules. *BIT Numerical Mathematics*, v. 46, n. 1, p. 195–202, 2006. ISSN 00063835.

[28] GOLUB, G. H.; WELSCH, J. H. Calculation of Gauss Quadrature Rules. *Mathematics of Computation*, v. 23, n. 106, p. 221, 2006. ISSN 00255718.

[29] O'HARA, H.; SMITH, F. Error estimation in the clenshaw-curtis quadrature formula. *The Computer Journal. Section A / Section B*, v. 11, 08 1968.

[30] B., T. A. A.; LANDAU, E. *Handbuch der Lehre von der Verteilung der Primzahlen.* [S.l.: s.n.], 2007. 87 p. ISSN 00255572.

[31] BACHMANN, P. Analytische Zahlentheorie. *Encyklopädie der Mathematischen Wissenschaften mit Einschluss ihrer Anwendungen*, p. 636–674, 1904.

[32] DELP, E. J.; MITCHELL, O. R. Moment preserving quantization [signal processing]. *IEEE Transactions on Communications*, v. 39, n. 11, p. 1549–1558, 1991. ISSN 0090-6778. Disponível em: <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=111432>.

[33] DRISCOLL, T. A.; HALE, N.; TREFETHEN, L. N. *Chebfun Guide.* [S.l.: s.n.], 2014.

[34] WIDROW, B. Analysis of amplitude-quantized sampled-data systems. *Electrical Engineering*, v. 80, p. 450–450, 06 1961.

[35] PELGROM, M. *Analog-to-Digital Conversion.* [S.l.: s.n.], 2017.

[36] GLASER, A.; LIU, X.; ROKHLIN, V. A Fast Algorithm for the Calculation of the Roots of Special Functions. *SIAM Journal on Scientific Computing*, v. 29, n. 4, p. 1420–1438, 2007. ISSN 1064-8275.

[37] OLVER, S.; TOWNSEND, A. A fast and well-conditioned spectral method. v. 0, p. 24–29, 2012. Disponível em: <http://arxiv.org/abs/1202.1347>.

[38] BOGAERT, I.; MICHIELS, B.; FOSTIER, J. ${O}(1)$ Computation of Legendre Polynomials and Gauss–Legendre Nodes and Weights for Parallel Computing. *SIAM Journal on Scientific Computing*, v. 34, n. 3, p. C83–C101, 2012. ISSN 1064-8275.