Budapest University of Technology and Economics
Faculty of Electrical Engineering and Informatics
Department of Measurement and Information Systems

Maximum likelihood estimation of ADC parameters

M. Sc. thesis

Author
Tamás Virosztek

Supervisor
István Kollár

19th December 2013

MŰEGYETEM 1782

DIPLOMATERVEZÉSI FELADAT

Virosztek Tamás

szigorló villamosmérnökhallgató részére

Analóg-digitális átalakítók paramétereinek maximum likelihood becslése

Az analóg-digitális átalakítók paramétereinek definiálása és becslése sokak számára fontos. Fontos a gyártóknak, hogy tudják, mit ellenőrizzenek, mit javítsanak, hogyan mérjék magukat a konkurrenciához. Fontos a készülékgyártóknak, hogy tudják: mire számíthatnak,. mit ellenőrizzenek gyártás közben, illetve végbemérésкor. És fontos a felhasználóknak, akik ezek ismeretében dönthetnek a megvásárolt készülékek minőségéről, tudják mikor kell reklamálniuk, illetve hogy mire számítsanak.

A fentiek miatt jött létre az IEEE 1241-es szabvány, mely definiálja ezeket a paramétereket, leírja, hogyan kell (hogyan lehet) mérni ezeket, és milyen körülményeket kell biztosítani a megismételhető mérésekhez. Ennek fontos része a szinuszos mérőjelekkel történő teszt. Azért is fontos, hogy a mérési adatok feldolgozása mennyire pontosan történik, mert a gerjesztő-jelek paramétereinek közvetlen mérése nagyon nehéz, ezért a gerjesztőjelet a mért adatokra történő illesztéssel szokták megállapítani. Az adatokra való illesztés szabványban leírt módszere a legkisebb négyzetes eljárás. Ehhez képest a maximum likelihood becslés, melyet a szabvány nem definiál,, pontosabb eredményeket ígér, a számítási bonyolultság növekedése árán.

A hallgató feladatai a következők:

- Tekintse át és foglalja össze a szinuszos gerjesztőjel paramétereinek az A/D konverter kimenetén mért jelsorozatból elvégezhető maximum likelihood becslési módszerének tulajdonságait, hibáit, előnyeit-hátrányait.
- Készítsen el egy Matlab alapú toolboxot, mely képes összehasonlítóan elemezni és illusztrálni az LS és ML módszereket. Tegye ezt az interneten elérhetővé.
- Lehetőség szerint építsen be többféle minimalizálási módszert.
- Nemzetközi együttműködésben készítse elő a Matlab és LabView alapú teszt-eljárások standardizálását és összehasonlítását.
- Tegyen javaslatot arra, hogy az internet segítségével hogyan lehet a mért adatok alapján összehasonlítható elemzéseket végezni.

Tanszéki konzulens: Dr. Kollár István egyetemi tanár (MIT)

Külső konzulens: -

Budapest, 2013. március 14.

/ Dr. Jobbágy Ákos /
egyetemi tanár
tanszékvezető

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Méréstechnika és Információs Rendszerek Tanszék

1117 Budapest, Magyar Tudósok krt. 2. I. ép. I.E.444.
Telefon: 463-2057, Fax: 463-4112
http://www.mit.bme.hu • e-mail: mitadm@mit.bme.hu

Contents

HALLGATÓI NYILATKOZAT

Alulírott *Virosztek Tamás*, szigorló hallgató kijelentem, hogy ezt a diplomatervet meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző(k), cím, angol és magyar nyelvű tartalmi kivonat, készítés éve, konzulens(ek) neve) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálózatán keresztül (vagy autentikált felhasználók számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik. Dékáni engedéllyel titkosított diplomatervek esetén a dolgozat szövege csak 3 év eltelte után válik hozzáférhetővé.

Budapest, 2013. december 19.

Virosztek Tamás
hallgató

Abstract

Analog-to-digital conversion is essential in embedded systems, where perception of the physical environment and digital processing of signals is required simultaneously, and mostly in real-time. Test methods for analog-to-digital converters (ADCs) have been improved in parallel with the development of the ADC circuits. Several techniques are available to observe the static and dynamic behavior of the converters. Driving the ADC under test using a sinusoidal excitation signal, and fitting sine wave to the measurement record is a very important and meaningful test procedure. This method has been standardized, and appears in documents released by the IEEE and the IEC. However, it is possible to improve this technique, using more accurate estimation of the excitation signal. Maximum likelihood (ML) estimation provides better estimators, thus datasheet quantities of the ADC under test can be calculated with improved precision. This paper focuses on the practical realization of ML estimation for ADC testing: examines several challenges concerning the implementation of this method and proposes solutions for these problems. Investigates the properties of the ML estimators in comparison with the properties of the standardized least squares (LS) estimators. Finally, implementation of the ML estimation (and multiple other, standardized ADC test methods) in two different environments (MATLAB and LabVIEW) is presented.

Kivonat

Beágyazott rendszerekben, ahol a fizikai környezet érzékelése és a mért jelek digitális feldolgozása egyaránt kulcsfontosságú, az analóg-digitális átalakítás nélkülözhetetlen. Az A/D átalakítók és a tesztelésükre szolgáló eljárások egyaránt fejlődtek az elmúlt évtizedekben: a konverterek statikus és dinamikus tulajdonságainak vizsgálatára számos módszer áll rendelkezésre. A szinuszjellel gerjesztett átalakító által mért jelelakra történő szinuszillesztés nagyon fontos és sok információt adó A/D tesztelési eljárás: az IEC és az IEEE által kiadott szabványokban is megtalálható. Ez a módszer azonban továbbfejleszthető a mérőjel paramétereinek pontosabb becslésével. A maximum likelihood (ML) becslési eljárás pontosabb szinuszillesztést eredményez, mint a szabványosított eljárás, ezáltal a tesztelt A/D átalakító minőségi jellemzői még pontosabban kiszámíthatók. Jelen diplomaterv a ML becslési eljárás gyakorlati megvalósításának kérdéseivel foglalkozik: feltár számos, a módszer implementációjával kapcsolatban felmerülő problémát, és megoldási javaslatokat kínál ezekre. Ezen kívül megvizsgálja a ML becslők tulajdonságait, és összehasonlítja a szabványos eljárásokban használt legkisebb négyzetes hibát adó (*„least squares", LS*) becslők tulajdonságaival. Végezetül bemutatja a ML becslés (és több más, szabványos A/D tesztelési eljárás) implementációját MATLAB és LabVIEW környezetben.

Introduction

Phenomena of the real, physical environment can be described using continuous-time and continuous-amplitude functions. However, digital signal processing can only treat finite amounts of data. On the one hand, in data processing and storage devices only the samples of the continuous-time function can be handled. On the other hand, these samples can only take a finite set of different values. The aim of analog-to-digital conversion is to create a *sampled* and *quantized* signal (practically a vector of digitally represented numbers) that contains the most information concerning the analog signal.

This report focuses on testing of analog-to-digital converters (ADCs). To overview the entire field of A/D conversion is not even attempted. The field of ADC test methods is also too large to be the subject of an M. Sc. thesis. In this paper both architectures, typical errors, and test methods are itemized more or less briefly to locate the more detailed disquisition of the subsequent chapters. Nevertheless the main goal of this report is to focus on a dynamic ADC test method, the examination of ADCs using sinusoidal excitation and sine wave fitting. This technique can be improved using maximum likelihood (ML) parameter estimation. The predominant part of this report deals with the theoretical background and practical implementation of the ML method for ADC testing.

A brief summary of commonly used architectures and typical errors of these devices is provided in chapter 1. Chapter 2 introduces the currently available European an American standards concerning ADC testing. Chapter 3 narrows the scope to the test methods using sinusoidal excitation. Theoretical background of ML estimation and its specialization for ADC testing are described in chapter 4 and chapter 5, respectively. Chapter 6 deals with a special issue of the model used for ML estimation: the amount of noise. Chapter 7 enumerates the challenges that make implementation of ML method difficult, and provides the suggested solutions to answer them. The novel method is compared to the standardized sine parameter estimation method (the least squares fit) in chapter 8. Chapter 9 introduces the software implementation of sine wave fitting using ML estimation in two different platforms: MATLAB and LabVIEW. Chapter 10 provides the conclusions of this report, and reflects to the specification. Chapter 11 enumerates the questions related to this topic, that can designate the path of further research.

This M. Sc. thesis is based on a „TDK" report (TDK is the abbreviation of *Scientific Students' Associations Conference* in Hungarian) written by the author [27]. This document is a largely revised and significantly extended version of the „TDK" report. Nevertheless there are major content overlaps between the two documents. There are figures and sections

of text that are taken from the TDK report without any modification. In these cases amendment would not improve the quality of the paper. However, chapters (1, 10, 11, 7.2, and 8.3.2) are entirely new, and all other chapters contain major revisions.

The „state of art" is described in the first five chapters: this knowledge is available in the literature, the techniques itemized here are used in practice, nevertheless chapter 5 expounds fresh results: [21] has been published in 2010. The subsequent chapters contain the results of own research efforts: these are concerning the practical realization of ADC testing using ML estimation.

Chapter 1

Typical architectures and imperfections of analog-to-digital converters

There are numerous devices that realize quantization or sampling in very different ways. For example, *rotary encoders* convert phase position to digital codes. In this case the phase range between 0 and 2π is quantized as the disk of the encoder is divided into 2^N parts. Another very widely used device is the *stroboscope*. Using short pulses of flashing light, periodical movements can appear to be reversed, stopped, or slowed down. The basis of this phenomenon is sampling: as only samples of the position of a moving object are available, the spectator has to interpolate somehow these samples. This way, according to the well-known principles of sampling and interpolation, the movements reconstructed can be strange, interesting or entertaining. Also analog motion picture strips use sampling. As only frames (samples of the sight) are stored, relatively high frequency periodic movements (such as rotation of a car wheel) can be apparently static or reversed.

In electrical engineering, analog-to-digital conversion is restricted to convert electrical quantities to digitally represented numbers. Electronic analog-to-digital converters (ADCs) sample and quantize voltage versus time, or current versus time functions. In the followings, the term ADC will be used for the devices that convert a voltage versus time function to a stream of digital codes. The subsections below enumerate the most frequent electronic realizations of sampling and quantization. As sample-and-hold and quantization functionalities are usually separated in the circuit design, the presentation of these functional blocks is also separated in this chapter.

1.1 Sample-and-hold architectures

The sample-and-hold functionality can be achieved many ways: using very simple circuits provide solutions with low cost and low component demand, more complex, even very complex circuits provide sample-and-hold functionality with less imperfection, and naturally require more resources. Figure 1.1 shows a schematic figure of a simple sample-and-hold

unit [1].

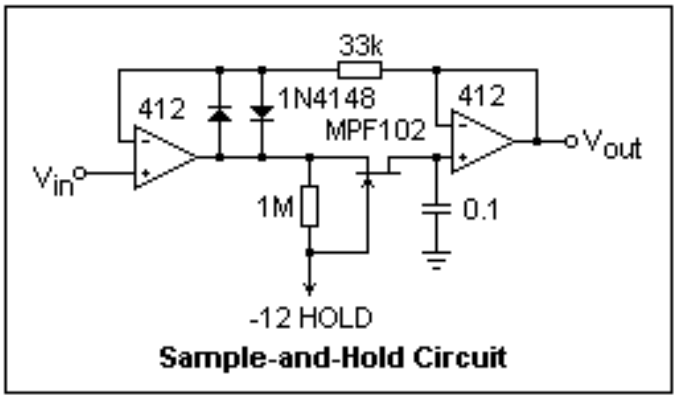

Figure 1.1: *A simple sample-and-hold architecture*

As it is visible on the figure, the sample-and-hold circuit has four main components: a high impedance input buffer to „hide" the capacitive load of the hold unit, a switch, a capacitor, and an output buffer to take over the load of the next stages of the converter. The electrical realization can be more complex to decrease imperfections of the circuit, however the principles and the basics of the architecture do not change. Figure 1.2 shows a more complex realization of sample-and-hold functionality [2].

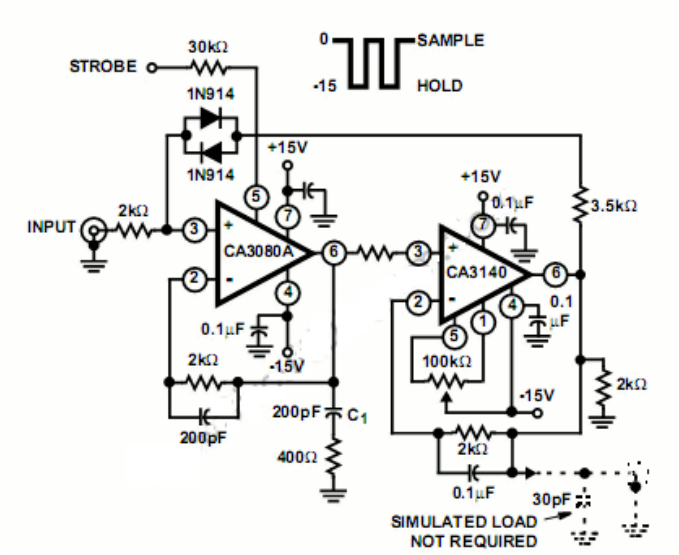

Figure 1.2: *A less simple sample-and-hold architecture*

The most common imperfections of a sample-and-hold (S&H) circuit are the followings [3]:

- **Aperture distortion:** this phenomenon appears when the switching time from sample mode to hold mode differs significantly from the switching time from hold mode to sample mode. This kind of error is caused by the nonlinear and asymmetrical behavior of the switching instant.
- **Aperture jitter:** in ideal case of equidistant sampling, the sampling time is equal

for each sample. However, every oscillator has a finite uncertainty, thus the sampling times differ from each other. Aperture jitter describes the standard deviation of the sampling times.

- **Track-mode distortion:** this imperfection is caused by the nonlinear U-I characteristic („input-dependent impedance") of the input buffer: the voltage drop depends nonlinearly on the input voltage, this way the signal on the output of the buffer (that charges the hold capacitor) is distorted.

- **Hold feed-through:** the parasitic capacitive and conductive coupling between the input and output provides stray effects when the input and the output shall be isolated (in hold mode). The output of the S&H circuit can be disturbed by the input signal through this parasitic coupling.

1.2 Quantizer architectures

The task of the quantizer circuit is to convert a stable voltage value provided by the sample-and-hold unit to a digital code. There are several fundamentally different quantizer architectures, the following sections will introduce the most commonly used ones.

1.2.1 Flash converters

The flash converter divides the full scale range (FSR) of the input using a *string resistor ladder*. The FSR is the voltage interval between V_{ref}^{-} and V_{ref}^{+} (in case of bipolar ADCs), and between 0 and V_{ref} (in case of unipolar ADCs). The code transition levels are the voltage values provided by the string resistor ladder. The output of the S&H circuit is directly and parallelly compared to these voltage values. The digital code is calculated from the results of the comparisons using a combinational logic. Figure 1.3 displays the architecture of a 3-bit flash quantizer circuit [4].

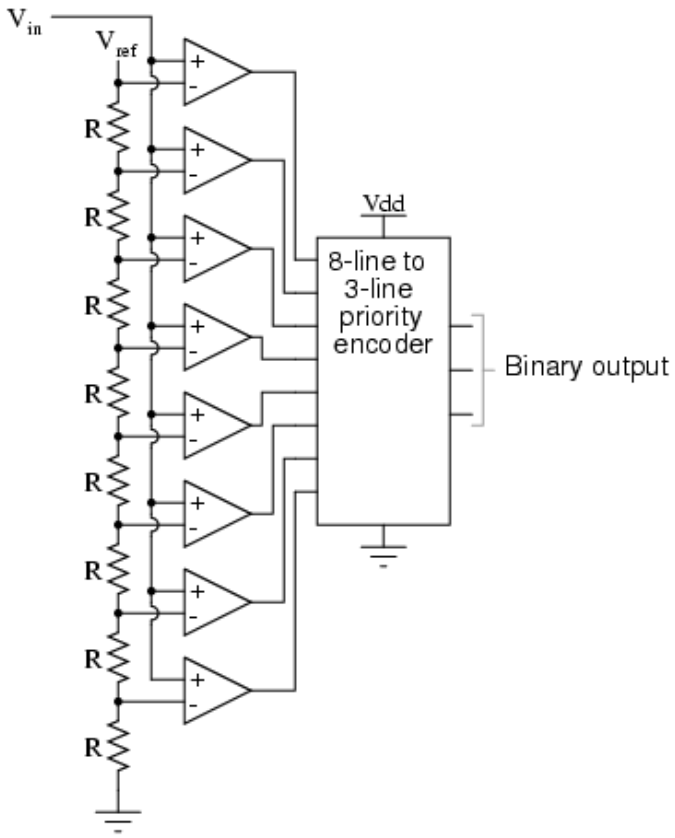

Figure 1.3: *Schematic diagram of a 3-bit flash quantizer*

An obvious disadvantage of flash quantizers is the high number of fixtures. To achieve N bits resolution, $2^N - 1$ comparators and 2^N resistors are required. A large amount of resistors with low uncertainty can be very expensive. The advantage of flash architecture is the speed of the conversion: the time consumption is mostly the propagation delay of the combinational logic.

1.2.2 Successive approximation quantizers

This architecture uses a digital-to-analog converter (DAC), a comparator and a logical circuit containing the successive approximation register (SAR). The digital code is created in multiple consecutive steps. First, the DAC provides a voltage that divides the FSR into two equivalent intervals. The input voltage is compared to this value: if greater, the most significant bit (MSB) of the code is 1, and successive approximation shall be continued in the higher half of the FSR. If smaller, MSB is 0, and successive approximation shall be continued in the lower half of the FSR. The lower bits are determined in the next steps the same way: the input voltage is compared to the middle of the actual voltage range, the code bit and the path of further approximation depends on the result of this comparison. Figure 1.4 provides a block scheme of a SAR quantizer architecture [5], and the time diagram of the successive approximation appears in figure 1.5 [6].

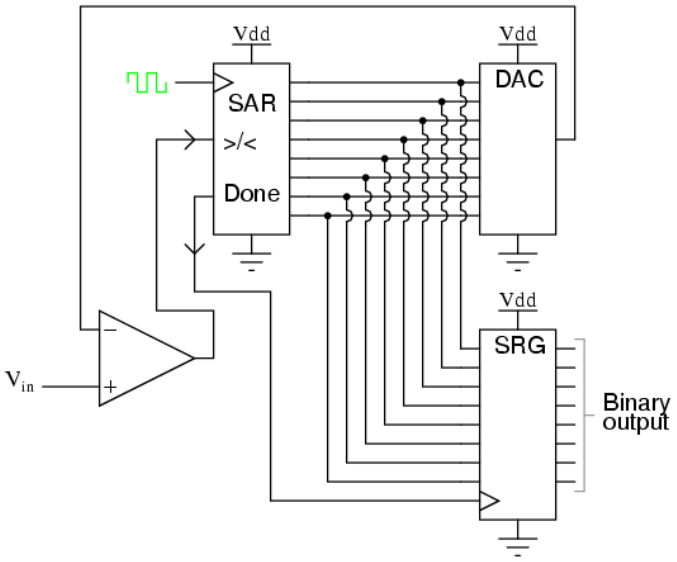

Figure 1.4: *Block scheme of a successive approximation quantizer*

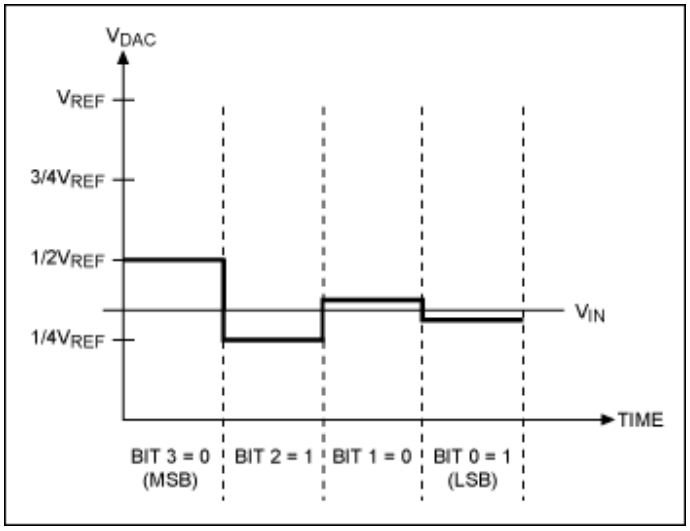

Figure 1.5: *Time diagram of successive approximation*

This method performs the quantization of a voltage value in multiple steps: each bit requires a step in the successive approximation process. Thus the conversion time of an N-bit SAR quantizer is $N \cdot T_{\mathrm{DAC+logic}}$, where $T_{\mathrm{DAC+logic}}$ denotes the time demand of the DAC to provide stable voltage output, and the propagation delay of the SAR logic circuit. These architectures can ensure typically 12..16 bit resolution and sample rate up to a few MS/s. Thus SAR ADCs are widely used in industrial applications, where low cost and relatively high performance is required.

1.2.3 Sigma-delta converters

Sigma-delta modulation, and its usage in analog-to-digital conversion is an entire topic of measurement technology. Hundreds of articles and several books are available regarding Σ-Δ conversion. [9] is a very thorough and precise handbook of this field. In this section

only the principles of sigma-delta modulation will be summarized briefly.

Figure 1.6 displays the block diagram of a Σ-Δ modulator used for analog-to-digital conversion [7].

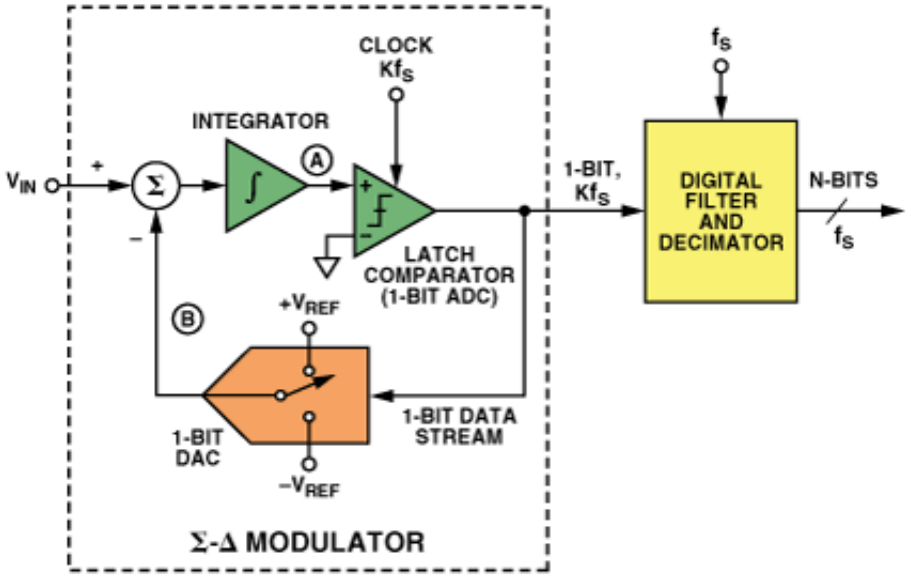

Figure 1.6: *Functional block diagram of the sigma-delta modulator*

A key movement of Σ-Δ modulation does not appear in the figure above: the oversampling of the input signal. While the sample-and-hold circuit provides the stable value of the sampled input, the sigma-delta modulator performs multiple analog-to-digital and digital-to-analog conversions. The number of these conversions applied on the same sample of the input signal is called *oversampling ratio* (OSR). This way the value of the input voltage appears pulse density modulated in the output of the comparator (i. e. the 1-bit ADC). This 1-bit oversampled signal is digitally low-pass filtered and then decimated by factor of the OSR. The output of the Σ-Δ ADC is the output of this digital filter. Figure 1.7 shows the time domain behavior of the sigma-delta modulator: the voltage versus time and code versus time functions of each part of the modulator are displayed using the same time axis ([8]).

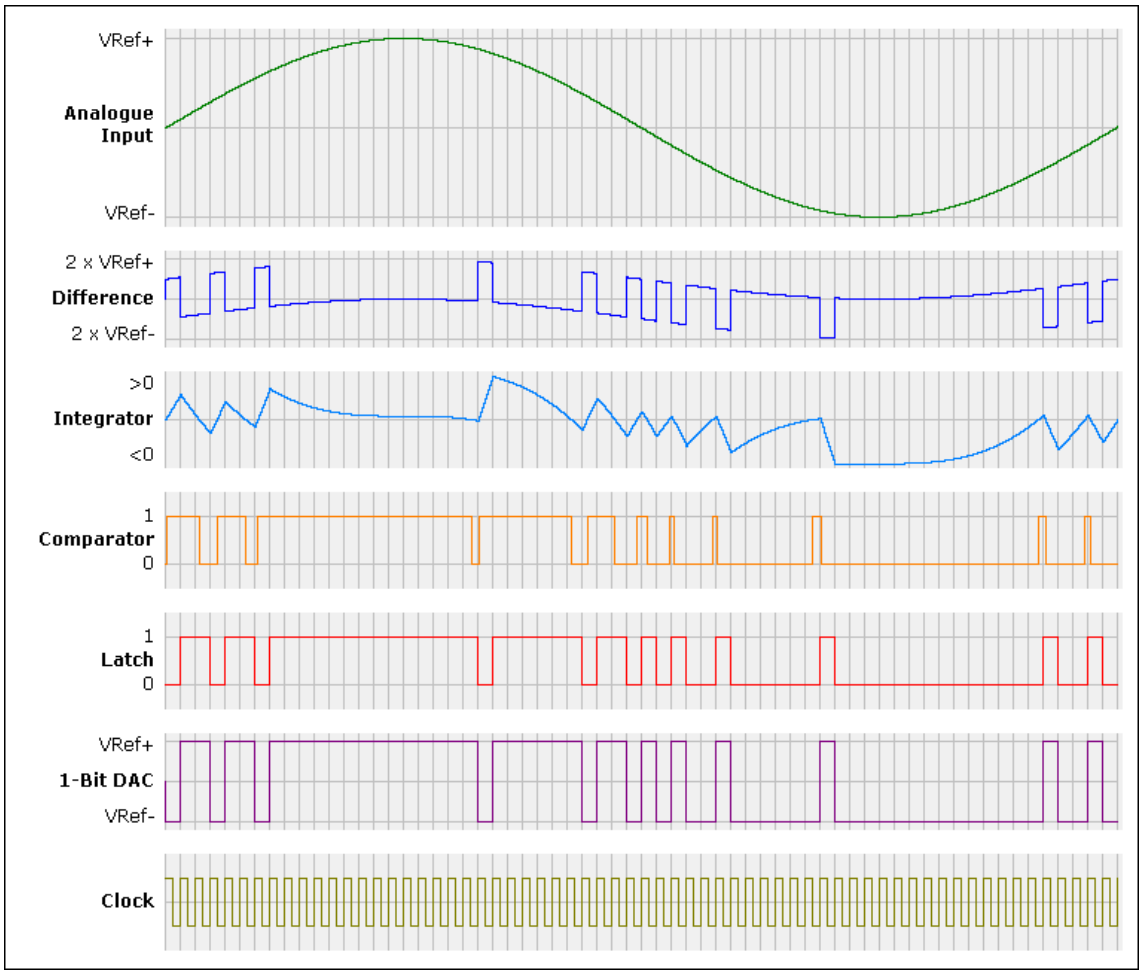

Figure 1.7: *Time domain behavior of the Σ-Δ modulator*

Observing the linear model of the sigma-delta modulator in the frequency domain (z-domain) helps to understand the *noise shaping* property of this architecture. In the linear model, the 1-bit ADC is replaced by a source of additive noise.

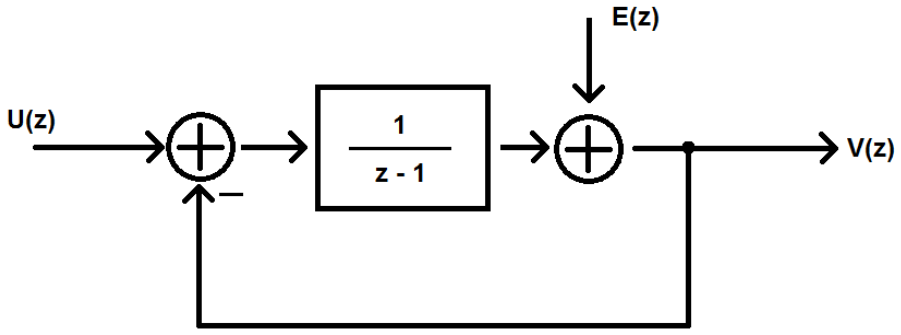

Figure 1.8: *Linear z-domain model of the Σ-Δ modulator*

This way the signal and the quantization noise are separated in the frequency domain:

$$V(z) = z^{-1}U(z) + (1 - z^{-1})E(Z) \tag{1.1}$$

While the signal has been delayed (z^{-1}), the quantization noise has been high-pass filtered: replacing z by $e^{j2\pi fT_s}$, the following equation describes the spectral properties of noise:

$$S_q(f) = (2\sin(\pi f T_s))^2 S_e(f) \tag{1.2}$$

where $S_e(f)$ denotes the power spectral density function of the original quantization noise (assumed to be uniform), and $S_q(f)$ denotes the power spectral density function of the shaped quantization noise on the output of the modulator. Thus the most of the quantization noise appears near the Nyquist frequency of the oversampled signal: the low-pass filtered and decimated signal contains much lower amount of quantization noise. Naturally more complicated sigma-delta architectures are also available, and used frequently: the internal DAC and ADC can be multi-bit devices, and multiple integrator loops can be used to achieve more powerful noise shaping.

Using sigma-delta converters is very attractive in audio signal processing: when the required sampling frequency is relatively low (e.g. 44.1 kHz, 48kHz or 96 kHz), the oversampling can be performed without taking excessive efforts (like using very high-speed devices).

1.2.4 Pipeline quantizer architectures

To increase the speed of data acquisition, the process of quantization can be split into multiple consecutive steps: this solution is called the pipeline architecture. Pipeline ADCs are usually *subranging*, but not necessarily. Subranging is a very commonly used solution to separate the steps of quantization. Each stage uses a highly linear few-bit quantizer and a few-bit highly linear DAC. The input voltage is quantized using the ADC, and the DAC provides the voltage value belonging to the code generated by the ADC. The difference of the input voltage and the output of the DAC (the residual voltage) is amplified, and the next stage performs the same operation on this residual voltage. The digital code is assembled by a combinational logic, using the values provided by the ADCs (figure 1.9, [7]).

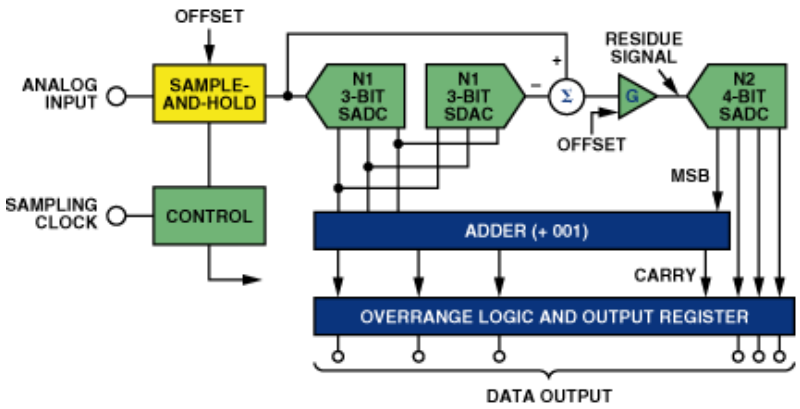

Figure 1.9: *Subranging ADC architecture*

1.2.5 Overview of architectures

The four different solutions itemized above are very commonly used ones to realize quantizers. Naturally other circuits are also used to perform analog-to-digital conversion, however they are not discussed in this report. Each architecture has advantages and disadvantages, specific errors and typical applications. Figure 1.10 overviews the different architectures, regarding typical resolution and sampling frequency values [7].

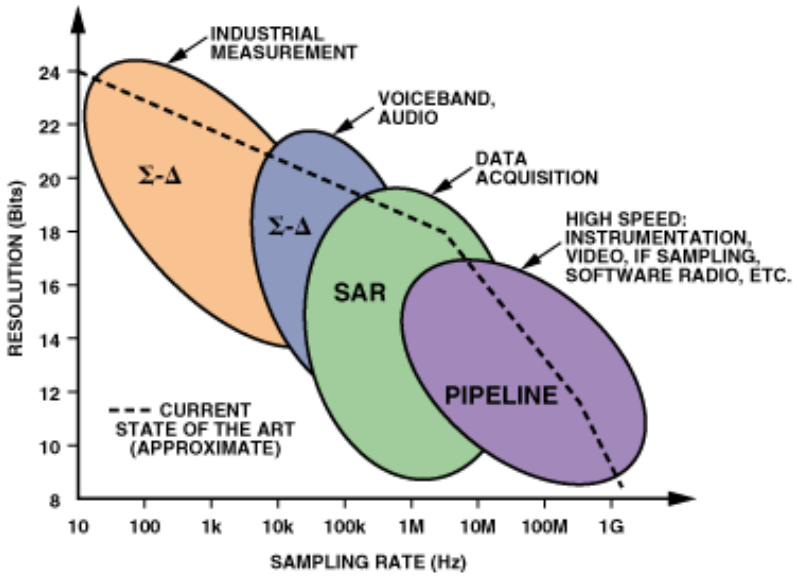

Figure 1.10: *Overview of ADC architectures*

1.3 Typical imperfections of analog-to-digital converters

The following paragraphs itemize the most important quantities that describe the non-ideal behavior of the ADCs. Some of these quantities characterize only the quantizer, others describe the behavior of the analog signal conditioning circuit, and some of them provides information about the entire data acquisition channel.

- **Input impedance**: the analog input of the device can be examined with DC excitation and AC excitation at various frequencies. Usage of a vector impedance meter is recommended. If measured impedances fit the model, input impedance can be expressed with the parallel combination of a capacitance and a resistance. Measurement of the reflection coefficient is also recommended using a time domain reflectometer (TDR). Input impedance can also be calculated from the value of the reflection coefficient, measured with a high quality cable, whose impedance is known accurately.

- **Crosstalk**: in case of multichannel devices crosstalk can be measured at any desired frequencies. This quantity is defined for each channel: to measure the multichannel crosstalk all *other* channels shall be driven with the maximum amplitude sine wave that can be applied. It is required to use common phase sine waves to maximize the effect of interference.

- **Common-mode rejection ratio (CMRR)**: This quantity is specified for signal analyzers using differential input. CMRR can be measured using a large amplitude sine wave that drives both terminals of the differential input. This way the excitation signal contains no differential mode component, but a large common mode component. In the case of ideal device, no signal should be recorded. The sine wave that appears in the waveform record is the effect of the common mode signal applied. Comparing the amplitude of the applied and measured sine wave, CMRR can be calculated.

- **Trigger delay and trigger jitter**: In digitizing oscilloscopes time delay between the trigger pulse, and the first sample recorded is also an important parameter. Trigger jitter is the deviation of the trigger delay. It can be estimated performing several trigger delay measurements. (Affined quantities aperture distortion and aperture jitter are detailed in section 1.1)

- **Out-of-range recovery**: input voltages outside the full scale range (FSR) of the digitizer can cause stray effects like saturation of an amplifier, or thermal effects because of the higher power dissipation. The recovery time of the device under test from the anomalous state to the state of normal operation can be measured using sine wave fitting. A large ($A < 120\%\text{FSR}$) and slow ($f < 0.05 f_{sampling}$) sine wave shall be recorded. When the signal leaves the treshold domain and returns to the full scale range, abnormal samples (samples not fitting to the sine wave) can be observed. Time delay between the return of the signal and the disappearance of the outlier samples can be estimated. The so called „mod T plot" of residuals can be very useful to observe out-of-range recovery.

- **Gain and offset**: The spread of the *real* full scale range can be expressed with the highest and lowest code transition voltage ($T[1]$ and $T[2^N - 1]$). However, it is more expressive to define quantities *offset* and *gain*. Offset denotes the difference between the endpoints of the theoretical and real full scale range, gain denotes the ratio of the theoretical and the real average code bin width.

- **Linearity**: The static transfer characteristic of an ADC is the set of code transition levels. Code transition level $T[k]$ is a DC voltage value, where the probability of output code $k-1$ and k are equal (50-50%). In case of linear ADCs distances between transition levels are equal, thus

$$T_{ideal}[k] = T[1] + Q \cdot (k - 1) \tag{1.3}$$

where Q denotes the average code bin width. *Integral nonlinearity* (INL) is the difference between the ideal and real code transition levels. Thus INL can be evaluated for each code transition:

$$\text{INL}[k] = \frac{T[k] - T_{ideal}[k]}{Q} \tag{1.4}$$

Integral nonlinearity appears on datasheets two different ways. In some cases a typical INL characteristic of the device is provided, INL[k] is plotted according to the transition levels. In other cases INL is given as the maximal absolute linearity error:

$$\mathrm{INL} = \max_{k=1}^{2^N-1} |\mathrm{INL}[k]| \tag{1.5}$$

- **Noise and distortion**: The most important quality parameters of an ADC are related to noise and distortion. Signal to noise ratio (SNR), signal to noise and distortion ratio (SINAD), and effective number of bits (ENOB) are both evaluated using the sine wave fit method in time domain. In these measurements a pure analog sine wave is recorded, then samples of a sine wave are fitted to the digital record. Sine wave fitting algorithms are detailed further, in section 2.3.

- **Step response**: Examining the step response is a very manifest test method for any device in signal processing. For ADC testing it is required that the imperfections of the test signal (transition duration, settling time and overshoot) shall not be greater than the 25 % of the imperfections expected from the device under test. By processing the record of the step response, quantities of the ADC can be calculated such as transition duration, overshoot, settling time, limits of slew rate, and many other parameters.

- **Frequency response parameters**: Examining the performance of the converter in the frequency domain is also important to provide useful information for the users. Frequency domain is usually swept from DC up to the *Nyquist frequency* of the device. To investigate the effects of aliasing, it also makes sense to use excitation signals over the Nyquist frequency. Bandwidth and gain flatness can be examined observing the amplitude response of the ADC in various frequencies. Most of the converters have no lower -3dB frequencies, as they are capable to measure DC voltages. The higher boundary of the passband is more important, and can be explained with multiple reasons. The limited bandwidth of the analog anti-aliasing filter (AAF) reduces the bandwidth of the entire device. Also slew rate limitations can be sources of lower amplitude response in higher frequencies. Gain flatness is a useful quantity to describe the worst case gain error in the passband:

$$E_G(f) = \frac{G(f) - G(f_{\mathrm{ref}})}{G(f_{\mathrm{ref}})} \tag{1.6}$$

$$E_G = \max_{i=1}^{N} |E_G(f_i)| \tag{1.7}$$

where f_{ref} is the reference frequency, where the amplitude response is considered ideal (in DC coupled devices f_{ref} is usually zero), and f_i denotes the frequency values, where the gain has been examined.

Chapter 2

Standardized methods for ADC testing

Analog-to-digital conversion is essential in electric engineering, since digital signal processing has been spread. In the last decades both ADC architectures and test methods have been improved simultaneously. As more and more devices are used worldwide, the demand for accurate and feasible test methods became very high. However, ADC testing standards are available only in Europe and in the USA. The American IEEE (Institute of Electrical and Electronics Engineers) and the European IEC (International Engineering Consortium) both have elaborated documents that clarify the terminology and test methods. Outside Europe and America the following cases are possible:

- one of the IEEE and IEC standards is used
- a mixture of the American and European standards is used
- other public documents, like [10] are used
- terminology and test methods differ from company to company.

Japanese engineer and researcher *Hauro Kobayashi* writes „In most Japanese companies ADC standards belong to engineers and there is no comprehensive document that can be used as a Bible of ADC standards and testing methods" ([11]). This article was published in 2001, when both standards IEEE-1241, IEEE-1057 and IEC-60748-4 were available.

While the usage of these documents is still confused in several cases, there are attempts to harmonize the American and European standards [12]. Nevertheless a unified and worldwide used standard seems to be a remote possibility only. Nowdays the best choice is to choose the one of the documents released, and to use the latest review of it consistently. The following sections describe the main traits of the internationally used standards, deal with the advantages and disadvantages of them, and counts the deficiencies of them.

2.1 IEEE standards

The American organization of electrical and electronics engineers (IEEE) provides two documents concerning ADC testing. Standard IEEE-1057 (last revised in 2007) deals with digitizing waveform recorders, thus its scope is wider than the scope of standard IEEE-1241 (last revised in 2010) that focuses on analog-to-digital conversion only.

2.1.1 Standard IEEE-1057

As this document was developed to define specifications and to describe test methods for each device that records electrical waveforms digitally, aspects outside the problem space of analog-to-digital conversion also appear in this standard. Digital oscilloscopes, digitizing waveform analyzers and recorders contain several other elements beyond ADCs, and these are also subject of examination. This standard deals with the entire path of data acquisition: transmission of the analog signal, signal conditioning, sampling and quantization. The main quantities defined here regarding the imperfections of a measurement channel are detailed in section 1.3.

2.1.2 Standard IEEE-1241

This document is focusing on terminology and test methods pronouncedly for analog-to-digital converters. Most of the definitions and procedures described here are also mentioned in the standard IEEE-1057. Nevertheless as 1057 reviews a larger area in the same spread, special issues of ADC testing are detailed only in the standard IEEE-1241. A very important part of this document is the description of sine wave fitting methods for dynamic testing. Sine wave fitting is fundamental to determine quantities like effective number of bits (ENOB) and signal to noise and distortion ratio (SINAD). Sine wave fitting in time domain is detailed in section 2.3.

2.2 IEC standards

In Europe, standards for electronic devices are provided by the International Electrotechnical Comission (IEC). Standard IEC-60748-4-3 [14] is regarding the dynamic criteria for analog-to-digital converters. This document has been developed by the subcommittee 47A (integrated circuits) of IEC technical committee 47 (semiconductor devices). First edition of the standard has been released in 2006.

2.2.1 Standard IEC-60748-4-3

This document is a brief summary of terminology and test methods for ADC testing. Contains only 36 pages instead of the American standards, that contain approximately 150 pages. However, parts of the standard IEC-60748-4 „Semiconductor devices - integrated circuits” [15] are referenced in this document. Terms and definitions for static behavior and characteristics of ADCs are described in Chapter II and Chapter III of the document

referred. In the fist part of standard IEC-60748-4-3 fundamental terms like coherency, equivalent sampling, code transition, linearity, SINAD, ENOB and others are defined very briefly. In the second part dynamic test methods like sinusoidal excitation, step response examination, and ramp signal excitation are detailed. In these sections definitions for calculated quantities such as THD or linearity error are provided. Mathematical derivations, restrictions on signal generators, and figures concerning the measurement setups appear in the annexes.

2.3 Fitting sine wave to a measurement record in least squares (LS) sense

2.3.1 LS fit using three sine wave parameters

Fitting samples of a sine wave to a measurement record can be performed minimizing the following cost function:

$$\mathrm{CF_{LS}} = \sum_{m=1}^{M} \left(y[m] - A\cos(2\pi f t_m) - B\sin(2\pi f t_m) - C\right)^2 \tag{2.1}$$

where M denotes the length of the record, t_m is the sampling time of the m^{th} sample, A is the cosine coefficient, B is the sine coefficient, and C is the DC component, and f is the frequency of the of the fitted sine wave. This cost function shall be minimized with respect to three parameters, A, B and C. The frequency of the signal is fixed. This least squares problem is linear in parameters, solution can be calculated in one step with simple matrix operations:

$$\mathbf{p} = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}\mathbf{y} \tag{2.2}$$

where

$$\mathbf{p} = \begin{bmatrix} A \\ B \\ C \end{bmatrix} \tag{2.3}$$

and

$$\mathbf{D} = \begin{bmatrix} \cos(2\pi f t_1) & \sin(2\pi f t_1) & 1 \\ \cos(2\pi f t_2) & \sin(2\pi f t_2) & 1 \\ \vdots & \vdots & \vdots \\ \cos(2\pi f t_M) & \sin(2\pi f t_M) & 1 \end{bmatrix} \tag{2.4}$$

and $\mathbf{y}$ is the measurement record:

$$\mathbf{y} = \begin{bmatrix} y[1] \\ y[2] \\ \vdots \\ y[M] \end{bmatrix} \tag{2.5}$$

2.3.2 LS fit using four sine wave parameters

The minimization of the cost function is more complicated, when the frequency of the signal is also a parameter to estimate. In this case an iterative solution can be performed: an initial frequency estimator f_0 shall be computed, and initial estimators for A_0, B_0 and C_0 shall be determined using three-parameter LS fit with given frequency f_0. Then four parameters A_i, B_i, C_i and Δf_i are to be calculated with the following equation:

$$p_i = \begin{bmatrix} A_i \\ B_i \\ C_i \\ \Delta f_i \end{bmatrix} = (\mathbf{D_i}^T \mathbf{D_i})^{-1} \mathbf{D_i y} \tag{2.6}$$

where

$$\mathbf{D_i} = \begin{bmatrix} \cos(2\pi f_i t_1) & \sin(2\pi f_i t_1) & 1 & -A_i t_1 \sin(2\pi f_i t_1) + B_i t_1 \cos(2\pi f_i t_1) \\ \cos(2\pi f_i t_2) & \sin(2\pi f_i t_2) & 1 & -A_i t_2 \sin(2\pi f_i t_2) + B_i t_2 \cos(2\pi f_i t_2) \\ \vdots & \vdots & \vdots & \vdots \\ \cos(2\pi f_i t_M) & \sin(2\pi f_i t_M) & 1 & -A_i t_M \sin(2\pi f_i t_M) + B_i t_M \cos(2\pi f_i t_M) \end{bmatrix} \tag{2.7}$$

Then frequency shall be adjusted: $f_i = f_{i-1} + \Delta f_{i-1}$, and matrix $\mathbf{D_i}$ shall be calculated again using f_i. Iteration can be terminated, when the frequency step Δf reaches a boundary defined previously.

Chapter 3

ADC test methods using sinusoidal excitation

In chapter 2 international standards concerning terminology and test methods for ADCs have been introduced. In these documents specific quantities describing the quality of an ADC are strictly defined, nevertheless the methods to evaluate these parameters of the device under test are summarized briefly. A detailed overview of all commonly used ADC test methods would fill an other report: in the last decades hundreds of articles have been published in the field of ADC testing. There are several fundamentally different approaches, and each test method has multiple practical realization. Some techniques are to examine the static behavior and others are to investigate the dynamic behavior of the device under test. There are methods to evaluate the performance of the ADC in time domain and in frequency domain. The type of the excitation signal on the input is also an aspect of classification. Any type of analog signals can be used for ADC testing. Comparison of the excitation signal and the digital record always provides information about the converter. The main problem is to reconstruct the waveform of the analog signal: the input of the device under test can only be measured using an other digitizing waveform recorder, thus to test an ADC, it is required to have an other, significantly better ADC. This requirement is absolutely impractical, and rises theoretical questions about how to examine the better ADCs. The way to avoid this problem is to estimate the analog signal using the digital record. This process makes the type of excitation signal important. Waveforms that can be described with only few parameters can be estimated in practice. Signals like multi-sine waves or arbitrary periodic signals with numerous harmonic components are barely appropriate for ADC testing, because they have numerous parameters to estimate.

Sawtooth signal can be described with only three parameters (amplitude, frequency, DC component). It is also useful for histogram testing, because the probability density function (PDF) of this signal is uniform between the two extrema, and zero elsewhere. But it is impossible to generate accurate sawtooth signal because of its unlimited bandwidth, and it is difficult to examine the quality of the signal generated. Other broadband signals like triangle or square waves rise the same problem.

Exponential signals also can be used for ADC testing. These signals can also be described

with three parameters, and can be generated using simple RC circuits. Naturally there are there are problems with generating pure exponential signals [16], but imperfections of an exponential can be described sufficiently [17]. The framework of ADC testing with exponential stimulus is described in [18].

Choosing sine waves as excitation signal is very attractive. On the one hand with sine waves the device is tested at one given frequency, so frequency-dependent behavior of the converter can be observed easily performing multiple measurements at different frequencies. On the other hand the quality of the sine wave can be observed easily, using a spectrum analyzer. As generating a pure sine wave is a very common task in electrical engineering, devices with very good parameters are available for laboratories at affordable prices. According to the subject of this report, an introduction to ADC testing with sine waves will be presented in the following sections.

3.1 The measurement setup

Examining ADCs with sinusoidal signals does not require complex measurement setup. A sine wave generator provides the signal that drives the analog input, and the response of the device (the digital codes) are recorded, most frequently by a PC. The generator can be an analog circuit (based on oscillators, and analog filters) or can be an arbitrary waveform generator with digital-to-analog converter (DAC). In each case it is essential to examine the quality of the signal created: harmonic distortion, peak-to-average ratio (PAR) and noise variance are important quantities to determine whether the generator is appropriate for the measurement or not. Recording the digital codes does not require any effort in case of standalone converters. These are usually connected to the PC with a high speed interface (USB or Ethernet), and the producer provides software support for the device. Our measurements were performed using National Instruments devices, the data were acquired with the NI LabVIEV software. In case of ADCs integrated to a system (e.g. a DSP or a microcontroller), recording and transmitting the samples is a programming task to be solved. The block scheme of the measurement setup is on figure 3.1.

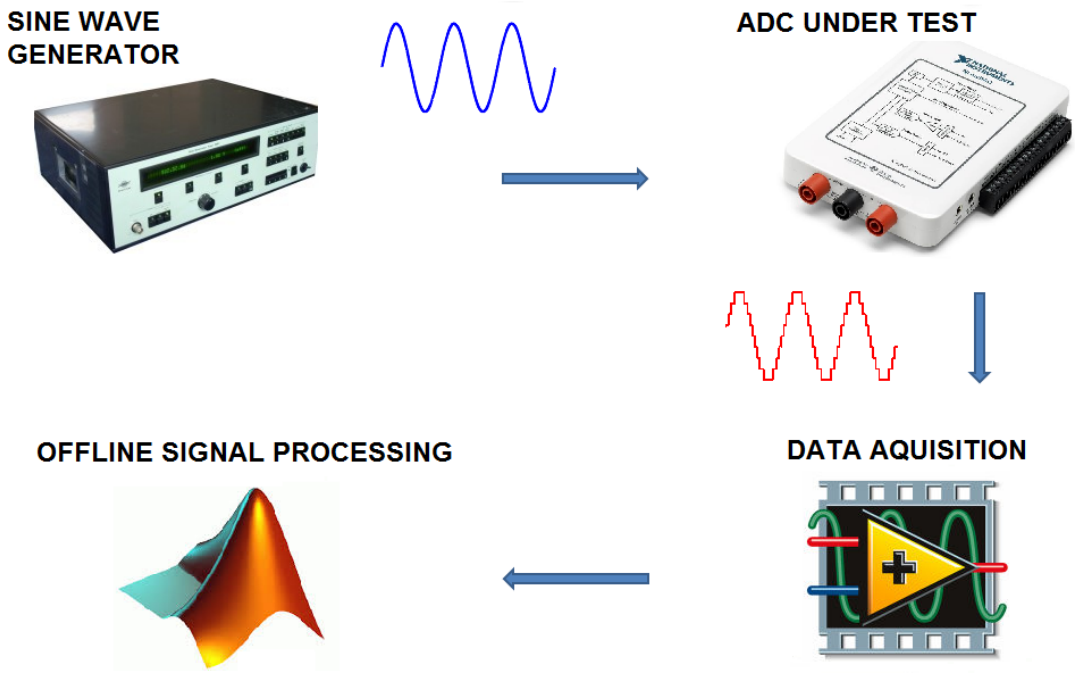

Figure 3.1: *Measurement setup for ADC testing with sine wave*

3.2 Quality of the test signal and environment

According to the simplicity of the measurement setup, there are only a few requirements to satisfy concerning the test environment. The pureness of the sine wave is a key issue in both static, both dynamic examination of the ADC. In histogram testing with sine wave the calculations are based on the PDF of the pure sine wave. Harmonic distortion changes the PDF of the real excitation signal, thus calculations with the PDF of pure sine wave can be misleading. In dynamic testing a single sine wave is fitted to the measurement record. If the record contains higher harmonic components, these appear in the difference of the record and the fitted sine wave. This difference is treated as the noise and distortion of the device under test. If the original excitation signal is distorted, it eventuates fake test results: harmonic distortion of the device appears to be higher than its real value. Presence of noise can also mislead the evaluation of the measurement. Histogram testing is surprisingly robust regarding the noise on the excitation signal: as noise changes the PDF of sine wave mostly near the extrema, effect of noise can be eliminated using high amplitude (120..150 % full scale) sine waves, that overdrive the converters [19]. Sine wave fitting methods are more sensitive to analog noise: similarly to harmonic distortion, noise in the measurement record is treated as noise of the ADC, so calculated parameters like ENOB and SINAD seem to be worse than their real value. On the other hand noise is an important signal parameter in maximum likelihood estimation, and numerical optimization cannot be performed without the presence of some noise on the analog side. Role and properties of noise are detailed in chapter 6.

Chapter 4

Framework and properties of maximum likelihood estimation

Maximum likelihood estimation is a very attractive method to estimate parameters of a model observing the output of the system. To perform estimation, a set of independent observations $(x_1, x_2, ..., x_n)$ are required. The parameters of the model (α) can be estimated via the so called *likelihood function*. To specify the likelihood function, it is necessary to express the *joint density function* of all observations:

$$f(x_1, x_2, ..., x_n|\alpha) = \prod_{i=1}^{n} f(x_i|\alpha) \tag{4.1}$$

where $f(x_i|\alpha)$ denotes the conditional probability density function of the observations with parameter α. The advantage of the independence of observations is obviously taken in this case. As we have fixed observations and an unknown parameter vector to estimate, the joint density function can be used in the following way:

$$\mathrm{L}(\alpha|x_1, x_2, ..., x_n) = f(x_1, x_2, ..., x_n|\alpha) = \prod_{i=1}^{n} f(x_i|\alpha) \tag{4.2}$$

The likelihood function (denoted by L) specifies the „agreement" of the parameter vector with the observations. Maximum likelihood estimator of α can be calculated maximizing the likelihood function:

$$\hat{\alpha}_{\mathrm{ML}} = \arg\max_{\alpha} \mathrm{L}(\alpha|x_1, x_2, ..., x_n) \tag{4.3}$$

To ease numerical computation, usually *log-likelihood* or *average log-likelihood* functions are optimized with respect to the parameters. Log-likelihood function is the natural logarithm of the likelihood function:

$$l = \ln \mathrm{L}(\alpha|x_1, x_2, ..., x_n) = \sum_{i=1}^{n} \ln f(x_i|\alpha) \tag{4.4}$$

Average log-likelihood function is normalized with the number of observations:

$$l_{\text{avg}} = \frac{1}{n} \ln \mathrm{L} = \frac{1}{n} \sum_{i=1}^{n} \ln f(x_i|\alpha) \tag{4.5}$$

Since logarithm function is strictly increasing, extrema of likelihood and log-likelihood functions are identical. However a sum can be derived more easily than a product, thus for optimization algorithms that require to calculate gradient or Hesse-matrix, it is strongly recommended to use the log-likelihood instead of the likelihood function.
Maximum likelihood estimator has the following properties [20]:

- **Consistency**: with sufficiently large number of observations, the *real* model parameters can be approximated with arbitrary precision. The ML estimator converges to the real value of the parameters.

$$\lim_{M \to \infty} \mathrm{P}[|\hat{\alpha}_{\text{ML}} - \alpha| > \epsilon] = 0 \tag{4.6}$$

  where M is the number of independent observations, and ϵ is an arbitrary low positive real number.

- **Efficiency**: when number of observations is tending to infinity, variance of the estimators reaches the Cramér-Rao bound.

- **Asymptotic normality**: the distribution of ML estimators is Gaussian with α mean, if number of samples tends to infinity.

Cramér-Rao bound, Fisher information, and covariance of estimators in case of ML estimation for ADC testing will be discussed in section 6.

Chapter 5

ML estimation of sine wave and ADC parameters

For maximum likelihood estimation of sine wave parameters for ADC testing, the following model has been developed [21]. The converter is described with a set of code transition levels. Transition level $T[k]$ is the value of the input voltage, that results code $k-1$ with probability of 50%, and code k with probability of 50% as well. The N-bit quantizer provides codes from 0 up to 2^N-1, and has 2^N-1 transition levels. The reduced full scale of the converter is the voltage range between the lowest and the highest transition levels ($T[1]$ and $T[2^N-1]$ respectively). Voltage values above the highest transition level result code 2^N-1 and voltages below the lowest transition level result code 0. Quantization can be described with a function $q(x)$ where

$$q(x) = \begin{cases} 0, & \text{if } x < T[1] \\ m, & \text{if } T[m] < x < T[m+1] \\ 2^N-1, & \text{if } x > T[2^N-1] \end{cases} \tag{5.1}$$

The sinusoidal excitation signal can be described with four parameters:

$$x(t) = A\cos(2\pi ft) + B\sin(2\pi ft) + C \tag{5.2}$$

where A is the cosine coefficient, B is the sine coefficient, and C denotes the DC component of the signal. The frequency of the sine wave is denoted with f. The electronic noise of the devices, and the imperfections of the measurement environment are modeled by additional noise on the excitation signal. Multiple noise models can be suitable for ML estimation, these will be itemized in chapter 6. The most manifest idea is to assume Gaussian noise with zero mean and σ standard deviation. Let $n(t)$ denote the realization of the additive noise. In this model the spectrum of the noise is white, so $n(\tau_1)$ and $n(\tau_2)$ are independent, if $\tau_1 \neq \tau_2$.

This noisy sine wave is quantized and sampled (the sequence is interchangeable), thus the output of the ADC can be described this way:

$$y(k) = q(x(t_k) + n(t_k)) \tag{5.3}$$

where t_k denotes the k_{th} sampling time moment (k = 1..M).
The parameters of the model to be estimated are the followings:

- The code transition levels of the quantizer: $T[1], T[2], ..., T[2^N - 1]$
- The cosine coefficient of the sine wave: A
- The sine coefficient of the sine wave: B
- The DC component of the sine wave: C
- The frequency of the sine wave: f
- The standard deviation of noise on the excitation signal: σ

As uniform sampling is assumed (effects of incidental non-ideal sampling are not considered in this model), the frequency of the sine wave can be described using the angular frequency normalized to the sampling frequency:

$$\theta = \omega T_s = 2\pi \frac{f}{f_s} \tag{5.4}$$

where T_s is the sampling time, and f_s denotes the sampling frequency. Thus the parameter vector to be estimated is the following:

$$\mathbf{p} = \begin{bmatrix} A \\ B \\ C \\ \theta \\ \sigma \\ T[1] \\ T[2] \\ \vdots \\ T[2^N - 2] \\ T[2^N - 1] \end{bmatrix} \tag{5.5}$$

To express the likelihood of the parameters, it is necessary introduce a vector of discrete random variables, denoted by $\mathbf{Y}$. Value $Y(k)$ belongs to the k^{th} sample of the measurement record and can achieve 2^N values: it can be any of the output codes of the ADC form 0 to $2^N - 1$ with a given probability. These probabilities can be described using the error function:

$$\mathrm{erf}(x) = \frac{2}{\pi} \int_0^x e^{-z} \mathrm{d}z \tag{5.6}$$

$$\mathrm{P}(Y(k)=0)=\frac{1}{2}\left[\mathrm{erf}\left(\frac{T[1]-x(t_k)}{\sigma\sqrt{(2)}}\right)+1\right] \tag{5.7}$$

$$\mathrm{P}(Y(k)=2^N-1)=\frac{1}{2}\left[1-\mathrm{erf}\left(\frac{T[2^N-1]-x(t_k)}{\sigma\sqrt{(2)}}\right)\right] \tag{5.8}$$

$$\mathrm{P}(Y(k)=l)=\frac{1}{2}\left[\mathrm{erf}\left(\frac{T[l+1]-x(t_k)}{\sigma\sqrt{(2)}}\right)-\mathrm{erf}\left(\frac{T[l]-x(t_k)}{\sigma\sqrt{(2)}}\right)\right] \tag{5.9}$$

where l = 1..2^N-2

To avoid the usage of three different cases, it is useful to define two „virtual" transition levels of the ADC: $T[0]=-\infty$ and $T[2^N]=+\infty$. This way the value of $Y(k)$ can be expressed in one equation:

$$\mathrm{P}(Y(k)=l)=\frac{1}{2}\left[\mathrm{erf}\left(\frac{T[l+1]-x(t_k)}{\sigma\sqrt{(2)}}\right)-\mathrm{erf}\left(\frac{T[l]-x(t_k)}{\sigma\sqrt{(2)}}\right)\right] \tag{5.10}$$

where l = 0..2^N-1

The likelihood function for the entire measurement is:

$$\mathrm{L}(\mathbf{p})=\prod_{k=1}^{M}\mathrm{P}(Y(k)=y(k)) \tag{5.11}$$

where $y(k)$ is the k^{th} sample of the digital record. Merging the equations above, one can express the likelihood function this way:

$$\mathrm{L}(\mathbf{p})=\prod_{k=1}^{M}\frac{1}{2}\left[\mathrm{erf}\left(\frac{T[y(k)+1]-x(t_k)}{\sigma\sqrt{(2)}}\right)-\mathrm{erf}\left(\frac{T[y(k)]-x(t_k)}{\sigma\sqrt{(2)}}\right)\right] \tag{5.12}$$

For computations, it is feasible to define a cost function, which is the negative log-likelihood function:

$$\mathrm{CF}(\mathbf{p})=-\ln\mathrm{L}(\mathbf{p}) \tag{5.13}$$

Chapter 6

Examination of measurement noise in ML estimation

6.1 Modeling the measurement noise in ADC testing

There are two requirements regarding the noise model used for ML estimation in ADC testing. On the one hand this model must be compatible with the physical phenomena that appear in electronic devices, on the other hand it must be treatable in the mathematical model. In ML estimation for ADC testing, the model is a source of additive noise superimposed on the pure excitation signal. This source replaces the noise of the original signal and the noise of the electronic devices in the ADC. This model complex enough to be absolutely useful to estimate signal parameters from measurement records, and simple enough not to rise unsolvable mathematical and numerical problems.

6.1.1 Gaussian noise model

In this model the noise is assumed to be Gaussian with $\mu = 0$ mean and σ standard deviation. Thus the probability density function (PDF) of the noise is

$$f_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{6.1}$$

Assuming that the electronic noise has many independent sources, and the resultant noise is the sum of many independent random variables, the Gaussian noise model is very attractive. Mathematical properties of this distribution are also suitable: the PDF can be derived any times anywhere, thus calculating partial derivatives of the cost function with respect to the parameters can be performed. To validate this model, real PDF of noise can be estimated taking long measurements with zero excitation, and creating a histogram. These results show that Gaussian model is a good approximation, however real measurement noise differs a bit from exact Gaussian distribution. Figure 6.1 shows a histogram of 1 million measured samples of noise.

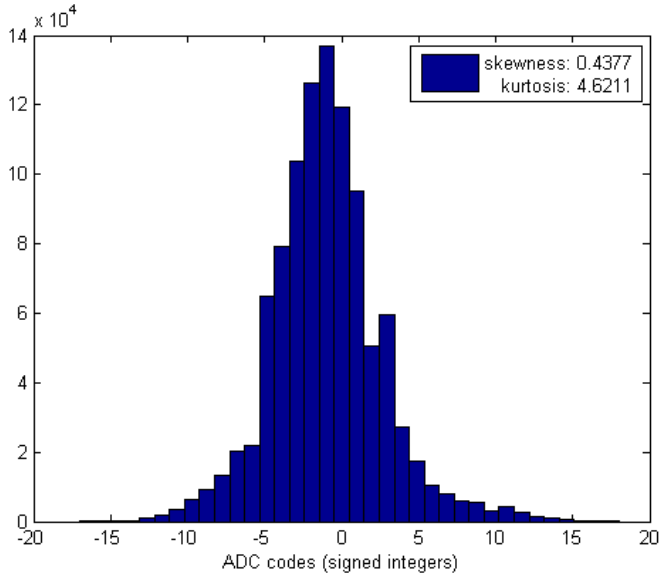

Figure 6.1: *Histogram of noise using 1 million samples*

The distribution is not symmetric (skewness is nearly 0.5), and the kurtosis is more than 150% of the kurtosis of normal distribution. To model the more outlier-prone distribution of noise, other noise models can be considered.

6.1.2 Laplace noise model

The noise can also be modeled with a random variable following the Laplace-distribution with $\mu = 0$ mean and λ scale parameter. Thus the PDF of the noise is

$$f(x) = \frac{1}{2}\lambda e^{-\lambda|x-\mu|} \tag{6.2}$$

This distribution assumes more outliers, and has an other desirable numerical property: the cumulative distribution function (CDF) can be evaluated using simple exponential calculations:

$$F(x) = \int_{-\infty}^{x} f(u)\mathrm{d}u = \begin{cases} \frac{1}{2}e^{\lambda(x-\mu)}, & \text{if } x < \mu \\ 1 - \frac{1}{2}e^{-\lambda(x-\mu)}, & \text{if } x \geq \mu \end{cases} \tag{6.3}$$

This fact can be important regarding computation time: using Gaussian noise model, the the error function is evaluated very numerous times (at least two times for each sample of the record). Evaluating a simple exponential is significantly faster than evaluating the error function that only can be approximated using numerical methods.
The derivatives of the PDF exist anywhere except for the 0. Thus the cost function can be derived partially with respect to the parameters except for those unlikely situations, when a sample of the pure sine wave in the model is equal with the value of a transition level. In these cases PDF cannot be derived, thus partial derivation of cost function cannot be performed.

6.1.3 Spectral distribution of noise

Maximum likelihood estimation requires independent observations on the output of a system to be estimated. To fulfil this requirement, samples of additive noise shall be independent at any sampling frequency. Thus spectral distribution of noise must be uniform. Examining long measurement records of noise show that white noise model is a very good approximation of the real noise spectrum. Figure 6.2 displays the amplitude spectrum of a noise measurement record containing 2 million samples. The sampling frequency is $f_s = 200$ kHz, thus the frequency resolution is $\Delta f = 0.1$ Hz. Some minor peaks appear near 20 kHz: these indicate the electromagnetic pollution of switching-mode power supplies. The emission of power lines can also be detected at 50 Hz, however these tiny imperfections do not question the validity of the white noise model.

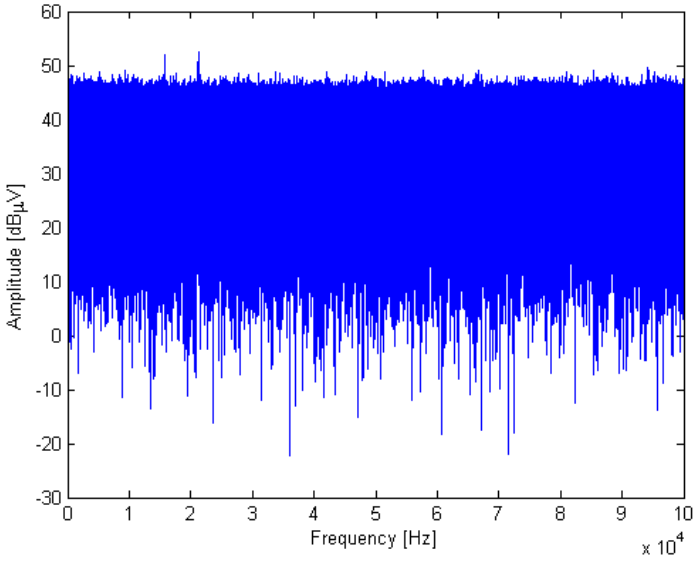

Figure 6.2: *Spectrum of measurement noise*

6.2 Role of noise in the optimization

As it was mentioned above, noise is a special parameter of the likelihood function. On the one hand σ is one of the parameters to be estimated: the $\hat{\sigma}_{\mathrm{ML}}$ estimator provides the maximum likelihood estimation for the deviation of noise. On the other hand in case of low-noise measurements, σ is rather a tool for the optimization algorithm to find the extrema of the cost function numerically. This special behavior is detailed below.
As $L(\mathbf{p})$ is a product of probabilities, defined by the error function, the value of each probability determines the overall likelihood of the measurement. For the k_{th} sample $y(k)$, the probability of being between the two corresponding transition levels $T[y(k)]$ and $T[y(k)+1]$ is the integral of the PDF of Gaussian distribution between $T[y(k)]$ and $T[y(k)+1]$. The mean of this distribution is the k^{th} sample of the pure sine wave, described using parameters A, B, C and θ.

$$\mathrm{P}[T[y(k)] < x(k)+n(k) \leq T[y(k)+1]] = \frac{1}{2}\left[\mathrm{erf}\left(\frac{T[y(k)+1]-x(k)}{\sqrt{2}\sigma}\right) - \mathrm{erf}\left(\frac{T[y(k)]-x(k)}{\sqrt{2}\sigma}\right)\right] \tag{6.4}$$

where $x(k)$ denotes the k^{th} sample of the pure sine wave: $x(k) = A\cos(k\theta) + B\sin(k\theta) + C$, and $n(k)$ is the k^{th} sample of the Gaussian noise. With the same sine wave parameters and different noise variance, these probabilities can be very different. On figure 6.3 the sample of the pure sine wave is between transition levels $T[y(k)]$ and $T[y(k)+1]$. In case of low noise variance, only output code $y(k)$ is compatible with the sine wave parameters A, B, C and θ. Increasing parameter σ, the adjacent codes also become compatible, they can occur with a finite probability.

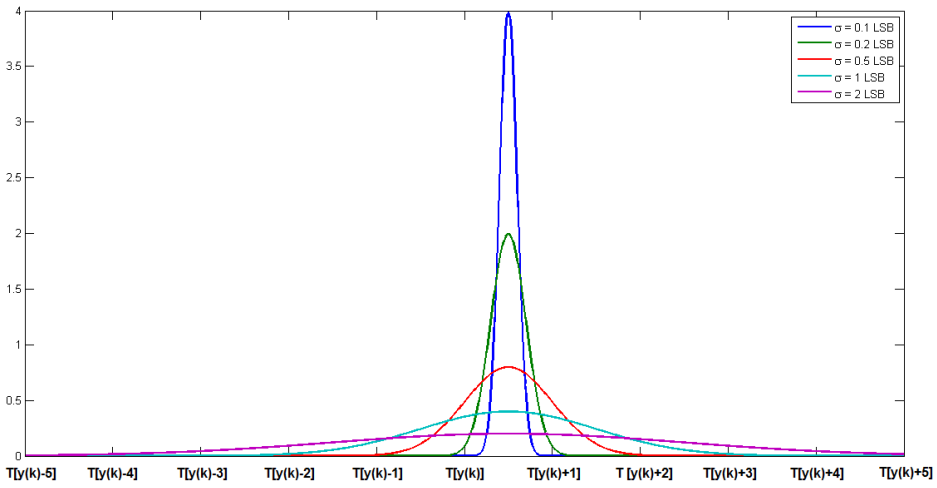

Figure 6.3: *Probability density of a noisy sample with different noise deviations*

This connection between noise and output codes can be reversed to likelihood: with a given measurement record, the set of compatible parameter vectors depends on the noise deviation. Permitting high σ, a wide range of parameter vectors becomes compatible with the measurement, but none of them is very likely. In case of low noise, only a narrow scale of parameter vectors is compatible, but their likelihood is high, can be nearly 1. When noise is arbitrary low ($\lim \sigma \to 0$), each sample can be totally compatible with the sine wave parameters (the likelihood of the parameters for that sample is 1) or can be totally incompatible (the likelihood of parameters for that sample is 0). This way the likelihood function can also be either 0 or 1 for a given parameter vector. In this case the first and second order partial derivatives (elements of the gradient and the Hess-matrix) are 0 or do not exist. Thus numerical optimization algorithms based on derivatives such as Gauss-Newton method or Levenberg-Marquardt method cannot optimize this likelihood function. The solution for this problem can be described in a few steps:

1. Calculate initial estimators for parameters A, B, C, and θ using four parameter sine wave fit in least squares sense.

2. Calculate estimators for transition levels using histogram of the record (be sure to fulfil requirements of histogram test).

3. Calculate initial estimator for noise using the quantized pure sine wave and the real measurement record:

$$\sigma_0 = \sqrt{\frac{1}{M-1}\sum_{k=1}^{M}(q(x_{\mathrm{LS}}(k)) - y(k))^2} \tag{6.5}$$

   where $x_{\mathrm{LS}}(k)$ is the k^{th} sample of sine wave assembled using LS estimators. If $\sigma_0 = 0$ (the quantized pure sine wave is identical with the measured sine wave), or too low, increase σ_0 artificially. Setting $\sigma_0 = 0.5$ LSB is a good decision according to our experience. Depending on the numerical evaluation of error function, it is recommended to increase σ_0 artificially if σ_0 is below 0.1 LSB.

4. Start optimization. The Levenberg-Marquardt method with scale factor (λ) adjusted appropriately at each iteration cycle, converges to the minimum of the cost function. If optimization fails according to numeric problems, the initial noise estimator might be too low. It shall be increased artificially as described in step 3.

The following figures trace the progress of optimization in case of low-noise measurement. The likelihood function is displayed with respect to two parameters: A and B. DC component and frequency are kept constant, and the deviation of noise changes in each step.

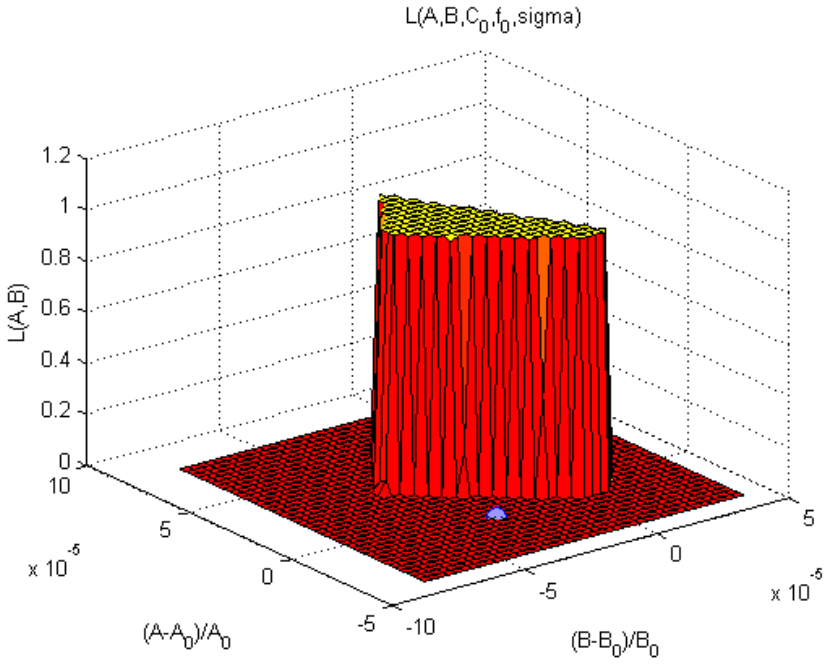

Figure 6.4: *Initial LS estimators are incompatible with the measurement: the likelihood is 0 and no derivatives can be calculated*

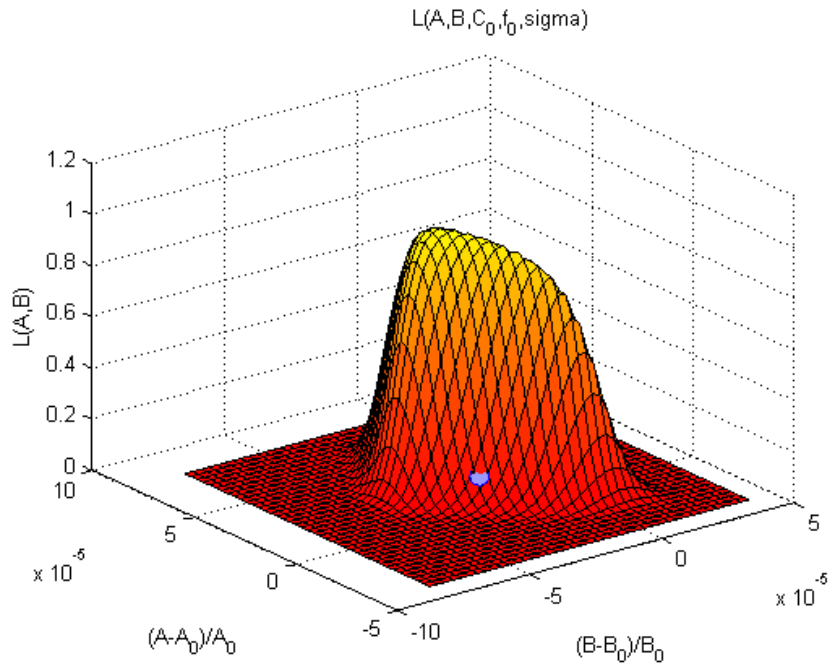

Figure 6.5: *Increasing σ smoothes the likelihood function: partial derivatives can be calculated, optimization can be initialized*

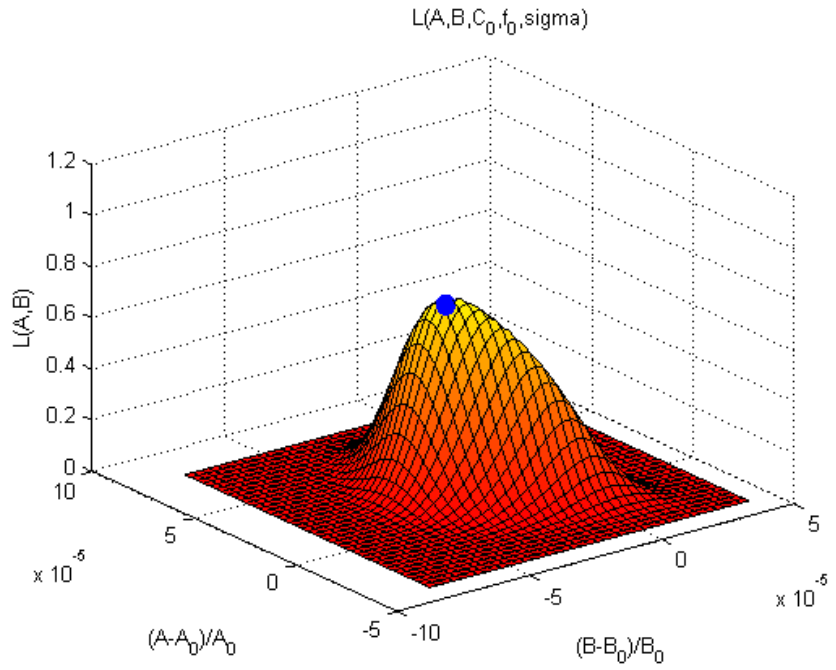

Figure 6.6: *The smoothed cost function can be optimized numerically: estimators find the extremum*

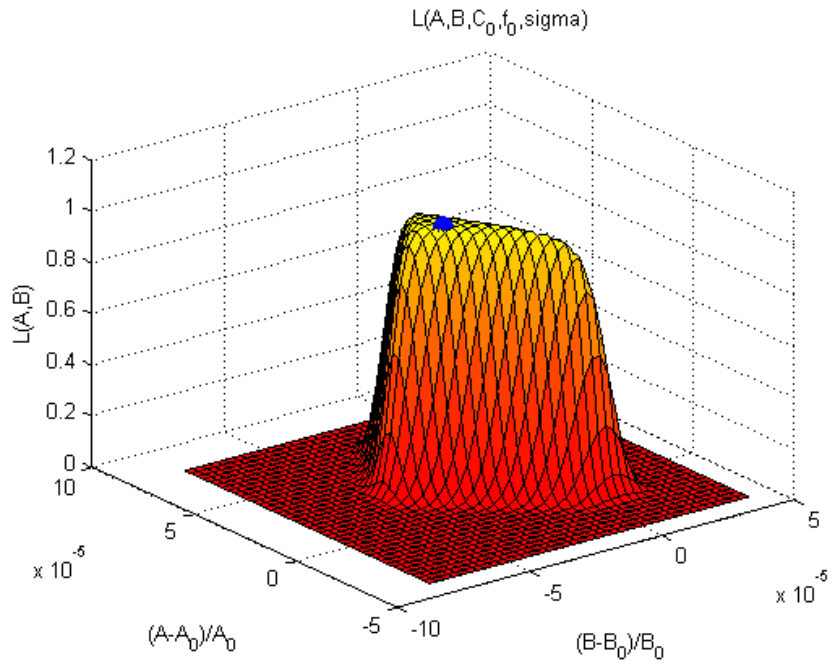

Figure 6.7: *Decreasing the noise deviation sharpens the likelihood function.*

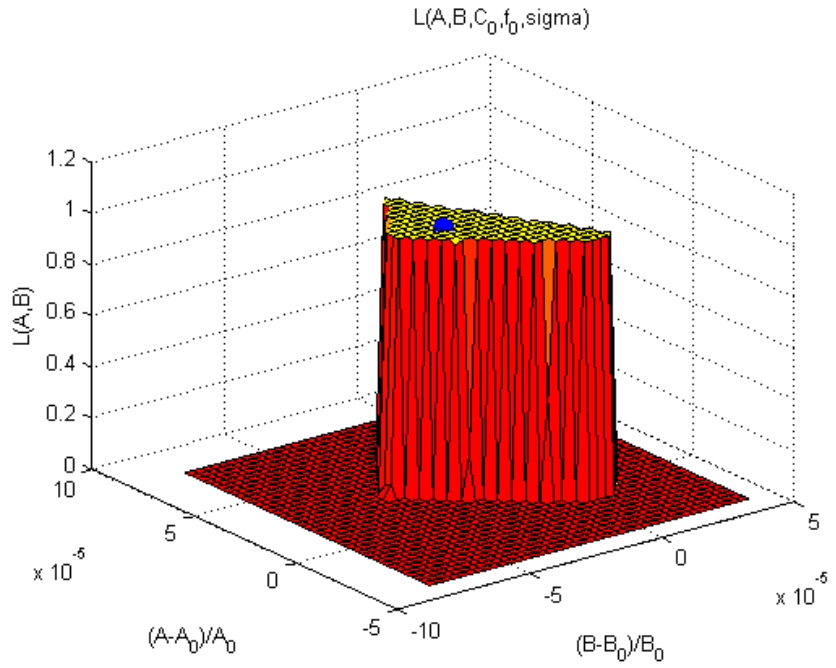

Figure 6.8: *Noise is decreased to arbitrary low value: the likelihood of the parameters reaches 1*

6.3 The effect of noise on the accuracy of estimators

The Cramér-Rao bound is the theoretical lower bound for the covariance of ML estimators. The covariance of estimators cannot be lower than the inverse of the Fisher-information. Fisher information is defined using second order partial derivatives of the log-likelihood

function:

$$\mathbf{I}(\mathbf{p}) = -\mathrm{E}\left[\frac{\partial^2 \ln L(\mathbf{p})}{\partial^2 \mathbf{p}}\right] \tag{6.6}$$

As the cost function is the negative logarithm of the likelihood function, the Fisher information can be expressed using the Hess-matrix of the cost function.

$$\mathbf{I}(\mathbf{p}) = \mathrm{E}\begin{bmatrix} \frac{\partial^2 \mathrm{CF}}{\partial^2 A} & \frac{\partial^2 \mathrm{CF}}{\partial A \partial B} & \frac{\partial^2 \mathrm{CF}}{\partial A \partial C} & \frac{\partial^2 \mathrm{CF}}{\partial A \partial \theta} & \frac{\partial^2 \mathrm{CF}}{\partial A \partial \sigma} \\ \frac{\partial^2 \mathrm{CF}}{\partial B \partial A} & \frac{\partial^2 \mathrm{CF}}{\partial^2 B} & \frac{\partial^2 \mathrm{CF}}{\partial B \partial C} & \frac{\partial^2 \mathrm{CF}}{\partial B \partial \theta} & \frac{\partial^2 \mathrm{CF}}{\partial B \partial \sigma} \\ \frac{\partial^2 \mathrm{CF}}{\partial C \partial A} & \frac{\partial^2 \mathrm{CF}}{\partial C \partial B} & \frac{\partial^2 \mathrm{CF}}{\partial^2 C} & \frac{\partial^2 \mathrm{CF}}{\partial C \partial \theta} & \frac{\partial^2 \mathrm{CF}}{\partial C \partial \sigma} \\ \frac{\partial^2 \mathrm{CF}}{\partial \theta \partial A} & \frac{\partial^2 \mathrm{CF}}{\partial \theta \partial B} & \frac{\partial^2 \mathrm{CF}}{\partial \theta \partial C} & \frac{\partial^2 \mathrm{CF}}{\partial^2 \theta} & \frac{\partial^2 \mathrm{CF}}{\partial \theta \partial \sigma} \\ \frac{\partial^2 \mathrm{CF}}{\partial \sigma \partial A} & \frac{\partial^2 \mathrm{CF}}{\partial \sigma \partial B} & \frac{\partial^2 \mathrm{CF}}{\partial \sigma \partial C} & \frac{\partial^2 \mathrm{CF}}{\partial \sigma \partial \theta} & \frac{\partial^2 \mathrm{CF}}{\partial^2 \sigma} \end{bmatrix} \tag{6.7}$$

As Hess-matrix of the cost function is calculated in each iteration cycle to perform Levenberg-Marquardt step, Cramer-Rao bound can be estimated using the Hess matrices calculated in the last iteration cycle. To achieve expected values of the second order derivatives, multiple (simulated) measurements performed with the same signal parameters shall be optimized, and average Hess matrices provides an estimator for the expected value.

$$\mathbf{cov}(\mathbf{p}) \geq \mathbf{I}^{-1}(\mathbf{p}) \tag{6.8}$$

6.3.1 Case of low noise

As it is visible on figure 6.8, in case of low noise, there is a set of parameter vectors with likelihood 1, and the likelihood of parameter vectors outside this set is 0. Each parameter vector with likelihood 1 is maximum likelihood estimator of the sine wave. Depending on the initial estimators and settings of the optimization algorithm, different parameter vectors with likelihood 1 can be reached. The Cramér-Rao bound cannot be calculated for this case: as the cost function is flat (constant 0) in the minima, the Hess-matrix is identically zero. Thus Fisher information matrix cannot be inverted and the theoretical lower bound for variance cannot be calculated.

6.3.2 Case of usual measurement noise

If measurement noise is high enough, shape of the likelihood function (and the cost function) is smoother: the curvature of the cost function near the minima can be calculated. This way it is possible to estimate the Fisher information (the expected value of the Hess-matrices), thus to approximate the Cramér-Rao bound.

Chapter 7

Implementation of ML estimation in practice: problems and solutions

7.1 Size of the parameter space

The most important challenge of the practical implementation is the problem concerning the parameter space. As it is described in chapter 5, the number of parameters strongly depends on the bit number of the conversion. As the parameter vector contains five signal parameters (A,B,C,θ, and σ) and $2^N - 1$ code transition levels, the length of parameter vector rises exponentially depending on the number of bits. As quantizers used in practice usually provide from 8 up to 24 bits resolution, the explosion of the parameter space rises serious challenges regarding computation time and numerical stability. To perform even the most simple numerical optimization algorithm, the negative gradient method, it is required to calculate the gradient of the cost function in each iteration:

$$\nabla \mathrm{CF}(\mathbf{p}) = \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial \mathbf{p}} = \begin{bmatrix} \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial A} \\ \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial B} \\ \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial C} \\ \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial \theta} \\ \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial \sigma} \\ \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial T[1]} \\ \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial T[2]} \\ \vdots \\ \frac{\partial \mathrm{CF}(\mathbf{p})}{\partial T[2^N-1]} \end{bmatrix} \tag{7.1}$$

The most obvious problem is to calculate $2^N + 4$ partial derivatives in each iteration cycle. However, this challenge can be answered allocating sufficient brute force (e. g. GPGPU-s) for the computation. The properties of partial derivatives with respect to the transition levels are more problematic. To examine this question, it is necessary to express the partial derivatives. The cost function is the negative log-likelihood function:

$$\mathrm{CF}(\mathbf{p}) = -\sum_{k=1}^{M} \ln \frac{1}{2} \left[\mathrm{erf}\left(\frac{T[y(k)+1] - x(k)}{\sqrt{2}\sigma} \right) - \mathrm{erf}\left(\frac{T[y(k) - x(k)]}{\sqrt{2}\sigma} \right) \right] \tag{7.2}$$

where $x(k)$ is the k^{th} sample of the pure sine wave:

$$x(k) = A\cos(k\theta) + B\sin(k\theta) + C \tag{7.3}$$

To ease overview of formulas, it is feasible to introduce the following notation:

$$\arg(k) = \mathrm{erf}\left(\frac{T[y(k)+1] - x(k)}{\sqrt{2}\sigma} \right) - \mathrm{erf}\left(\frac{T[y(k) - x(k)]}{\sqrt{2}\sigma} \right) \tag{7.4}$$

The first order partial derivatives of the cost function are expressed below:

$$\frac{\partial \mathrm{CF}}{\partial A} = -\sum_{k=1}^{M} \frac{1}{\arg(k)} \frac{2}{\sqrt{\pi}} \left(e^{-\left(\frac{T[y(k)]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{\cos(k\theta)}{\sqrt{2}\sigma} - e^{-\left(\frac{T[y(k)+1]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{\cos(k\theta)}{\sqrt{2}\sigma} \right) \tag{7.5}$$

$$\frac{\partial \mathrm{CF}}{\partial B} = -\sum_{k=1}^{M} \frac{1}{\arg(k)} \frac{2}{\sqrt{\pi}} \left(e^{-\left(\frac{T[y(k)]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{\sin(k\theta)}{\sqrt{2}\sigma} - e^{-\left(\frac{T[y(k)+1]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{\sin(k\theta)}{\sqrt{2}\sigma} \right) \tag{7.6}$$

$$\frac{\partial \mathrm{CF}}{\partial C} = -\sum_{k=1}^{M} \frac{1}{\arg(k)} \frac{2}{\sqrt{\pi}} \left(e^{-\left(\frac{T[y(k)]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{1}{\sqrt{2}\sigma} - e^{-\left(\frac{T[y(k)+1]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{1}{\sqrt{2}\sigma} \right) \tag{7.7}$$

$$\frac{\partial \mathrm{CF}}{\partial \theta} = -\sum_{k=1}^{M} \frac{1}{\arg(k)} \cdot \frac{\partial \arg(k)}{\partial \theta} \tag{7.8}$$

where

$$\frac{\partial \arg(k)}{\partial \theta} = \frac{2}{\sqrt{\pi}} \left(e^{-\left(\frac{T[y(k)+1]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{A\sin(k\theta)k - B\cos(k\theta)k}{\sqrt{2}\sigma} - e^{-\left(\frac{T[y(k)]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{A\sin(k\theta)k - B\cos(k\theta)k}{\sqrt{2}\sigma} \right) \tag{7.9}$$

$$\frac{\partial \mathrm{CF}}{\partial \sigma} = -\sum_{k=1}^{M} \frac{1}{\arg(k)} \cdot \frac{\partial \arg(k)}{\partial \sigma} \tag{7.10}$$

where

$$\frac{\partial \arg(k)}{\partial \sigma} = \frac{2}{\sqrt{\pi}} \left(e^{-\left(\frac{T[y(k)]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{T[y(k)] - x(k)}{\sqrt{2}\sigma^2} - e^{-\left(\frac{T[y(k)+1]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{T[y(k)+1] - x(k)}{\sqrt{2}\sigma^2} \right) \tag{7.11}$$

The case of the partial derivatives with respect to the transition levels is more complicated:

the numerical expression is simple, however the evaluation of this expression can be difficult depending on the measurement record.

$$\frac{\partial \mathrm{CF}}{\partial T[l]} = -\sum_{k=1}^{M} \frac{1}{\arg(k)} \cdot \frac{\partial \arg(k)}{\partial T[l]} \tag{7.12}$$

where

$$\frac{\partial \arg(k)}{\partial T[l]} = 0 \tag{7.13}$$

if $y(k) \neq l-1$ and $y(k) \neq l$. For the majority of k values (for the majority of the samples in the record), the elements of the sum are zero. Elements are nonzero only if $y(k) = l-1$, in this case

$$\frac{\partial \arg(k)}{\partial T[l]} = \frac{2}{\pi} e^{\left(\frac{T[l]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{1}{\sqrt{2}\sigma} \tag{7.14}$$

and if $y(k) = l$, in this case

$$\frac{\partial \arg(k)}{\partial T[l]} = -\frac{2}{\pi} e^{\left(\frac{T[l]-x(k)}{\sqrt{2}\sigma}\right)^2} \cdot \frac{1}{\sqrt{2}\sigma} \tag{7.15}$$

Summarizing the facts detailed above, optimizing the cost function in the entire parameter space is very problematic. The number of parameters increases exponentially depending on the bit number of the device under test, and partial derivatives with respect to the transition levels are very small, or even can be zero. Howewer, it is possible to find a satisfying approximate solution for the maximum likelihood problem.

7.1.1 Reduction of the parameter space

To perform this approximation, it is required to get transition level estimators using other methods, and optimizing the cost function with respect to the parameters that are dominantly determine the likelihood: the four signal parameters (A, B, C, θ), and the deviation of the noise (σ). Noise is a special parameter in ML estimation, properties and effects of noise are detailed in chapter 6. To estimate static transfer characteristic of the ADC, it is manifest to use histogram testing. It is important to declare that no further measurements are required: histogram test can be performed using the same measurement record. As probability density function (PDF) of a sine wave is well-known, it is possible to estimate integral nonlinearity (INL) and differential nonlinearity (DNL) of the device, using the formulas developed to evaluate histogram of a sinusoidal record [19]. This way histogram testing and sine wave fitting shall be computed using the same data: the samples of the sine wave recorded.

7.1.2 Appropriate estimation of transition levels

To perform histogram test properly, there are some requirements that are easy to describe in theory, and can barely be satisfied in practice. These requirements ensure that the

distribution of samples in the phase space is uniform between 0 and 2π.

- **Coherence**: a record is coherent, if integer number of sine wave periods are sampled. For M samples and T_s sampling time

$$M \cdot T_s = N \cdot T_0 \tag{7.16}$$

  where N is a positive integer, and T_0 is the reciprocal of the sine wave frequency.

- **Relatively prime condition**: the number of periods in a record (N) and the number of samples (M) shall be relatively primes. This condition ensures that each sample is in a unique phase position.

To fulfil these requirements, it is necessary to set the frequency of the sine wave generator appropriately. However, as both the ADC under test and the generator have frequency uncertainties, the record can be misleading. It is necessary to investigate coherence *after* the measurement, using a very accurate frequency estimator. This way, based on an estimator of the *real* f/f_s value, it is possible to discard samples of the record outside the coherent part, or to suggest other frequency settings on the sine wave generator, to achieve better phase distribution of samples. [22] provides detailed information about the framework of examination of coherence. Using proper measurement records, the transition levels can be estimated with any desired accuracy at a given confidence level using enough number of samples in a record. Formulas to calculate the required number of samples are described in [19]. As required record length can be long at higher bit numbers (e. g. for a 16-bit converter it is not superfluous to acquire multiple million samples), it is a feasible option to estimate the transition levels from the entire record, and to perform the ML sine wave fit using a shorter section. This procedure is useful to speed up computation. Behavior of estimators with increasing amount of data is examined in chapter 8.

7.2 Numerical optimization of the cost function

There are numerous methods and algorithms to find extrema of a multidimensional function. The literature of numerical recipes [23] largely exceeds the needs of this optimization problem. The cost function of the likelihood problem has been described in chapter 5. As most of the optimization algorithms use partial derivatives, it is important to consider that any order partial derivatives of this cost function can be calculated analytically. Approximation of these derivatives can also be calculated using finite differences, however this option is rather useful to check computations, finite differences do not provide more information than the analytically calculated values.

The numerical recipe chosen to solve this problem is the *Levenberg-Marquardt* method ([24], [25]). This recipe is scalable, uses the first and second order order partial derivatives, and provides fast convergence, if the scale factor of the step is adjusted well during the optimization. The initial parameter estimators ($\mathbf{p}_0$) are achieved via least squares sine wave fitting and histogram test. The Levenberg-Marquard step is the following:

$$\mathbf{p}_{k+1} = \mathbf{p}_k - (\mathbf{H} + \lambda \mathbf{I})^{-1} \cdot \left. \frac{\partial \mathrm{CF}}{\partial \mathbf{p}} \right|_{\mathbf{p}_k} \tag{7.17}$$

where $\mathbf{I}$ is the identity matrix, λ is the scale factor of the step and $\mathbf{H}$ is the Hess-matrix assembled using the second order partial derivatives:

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 \mathrm{CF}}{\partial^2 A} & \frac{\partial^2 \mathrm{CF}}{\partial A \partial B} & \frac{\partial^2 \mathrm{CF}}{\partial A \partial C} & \frac{\partial^2 \mathrm{CF}}{\partial A \partial \theta} & \frac{\partial^2 \mathrm{CF}}{\partial A \partial \sigma} \\ \frac{\partial^2 \mathrm{CF}}{\partial B \partial A} & \frac{\partial^2 \mathrm{CF}}{\partial^2 B} & \frac{\partial^2 \mathrm{CF}}{\partial B \partial C} & \frac{\partial^2 \mathrm{CF}}{\partial B \partial \theta} & \frac{\partial^2 \mathrm{CF}}{\partial B \partial \sigma} \\ \frac{\partial^2 \mathrm{CF}}{\partial C \partial A} & \frac{\partial^2 \mathrm{CF}}{\partial C \partial B} & \frac{\partial^2 \mathrm{CF}}{\partial^2 C} & \frac{\partial^2 \mathrm{CF}}{\partial C \partial \theta} & \frac{\partial^2 \mathrm{CF}}{\partial C \partial \sigma} \\ \frac{\partial^2 \mathrm{CF}}{\partial \theta \partial A} & \frac{\partial^2 \mathrm{CF}}{\partial \theta \partial B} & \frac{\partial^2 \mathrm{CF}}{\partial \theta \partial C} & \frac{\partial^2 \mathrm{CF}}{\partial^2 \theta} & \frac{\partial^2 \mathrm{CF}}{\partial \theta \partial \sigma} \\ \frac{\partial^2 \mathrm{CF}}{\partial \sigma \partial A} & \frac{\partial^2 \mathrm{CF}}{\partial \sigma \partial B} & \frac{\partial^2 \mathrm{CF}}{\partial \sigma \partial C} & \frac{\partial^2 \mathrm{CF}}{\partial \sigma \partial \theta} & \frac{\partial^2 \mathrm{CF}}{\partial^2 \sigma} \end{bmatrix} \tag{7.18}$$

The adjustment of the scale factor is the following: the initial value of λ is the greatest eigenvalue of the Hess-matrix, calculated at $\mathbf{p}_0$. If the Levenberg-Marquardt step achieves lower cost function than the previous, the optimization becomes „more courageous": λ decreases, thus the value of the Hess matrix becomes more dominant in the optimization. Low λ values are useful at the end of the optimization: nearby the extremum the cost function is approximately parabolic. If the step results higher cost function, or leads to an invalid parameter domain (e. g. a parameter vector, where $\sigma < 0$), the optimization becomes „more careful": λ increases, thus the direction of the step becomes more similar to the gradient, and the length of the step becomes lower. In my implementation, the value of λ increases and decreases by a factor of 10. For these cost functions this setting is seemingly appropriate: provides convergence, but careful enough: optimization has not been misled yet for numerous kinds of measurement records. ed yet for numerous kinds of measurement records.

7.2.1 Termination criteria

It is also important to terminate the iteration at an appropriate point: naturally this point depends on the requirements concerning the accuracy of the estimators. Some usual viewpoint of termination are itemized below:

- **Maximal number of iterations**: iteration terminates after a specified number of iterations.
- **Maximal number of cost function evaluations**: iteration terminates after a specified number of evaluation of the cost function (and its partial derivatives).
- **Termination tolerance on cost function**: if the cost function is smooth enough (changes less than a specified value), iteration terminates.
- **Termination tolerance on parameters**: if the parameters change less than a specified value, iteration terminates. This criterium usually compares the *euclidean distance* between the parameter vectors of two consecutive steps to a given scalar value. In case of parameters with different physical dimensions (e.g. voltage and frequency), scaling becomes very important.

In the implementations (described in chapter 9) termination criteria are handled differently: in the MATLAB toolbox the user can specify the values (default values are available), in the LabVIEW implementation these are hard-coded yet: user can only perform optimization using the default settings.

Chapter 8

Experimental comparison of ML and LS estimators for ADC and sine wave parameters

General properties of maximum likelihood estimators are detailed in chapter 4. In the following sections experimental results are provided to observe behavior of ML estimators used for ADC testing. These results are displayed in comparison: properties of the ML estimators are compared to the properties of the least squares estimators (that are used in the standardized methods).

8.1 Method of observation

To examine the quality of estimators it is required to know the exact value of the given parameter. Investigating estimator properties such as consistency or variance is impossible without the theoretical value of the parameter estimated. In case of real measurements only other estimators are available regarding the signal parameters. Thus simulated measurements shall be performed to examine behavior of estimators. The framework of investigation contains the following steps:

1. Chose a fix set of signal parameters. These will be used in the steps detailed above.
2. Perform multiple simulated measurements with the same signal parameters and the same *amount* of noise. As the realization of noise is different in each case, but signal parameters do not change, variance of estimators can be observed via these repeated measurements.
3. Create different sets of measurements, varying the number of sample, amount of noise, resolution, etc.
4. Process the simulated measurement record and store the estimators achieved.
5. Compare the calculated estimators to the real parameters: observe variance, consistency, etc.

8.2 The simulated measurements

Instead of using simulated measurement data created by myself, I have processed data provided by Ing. Jozef Lipták, PhD student of the Technical University of Kosice. In these simulations the device under test is an 8-bit nonlinear ADC. The static transfer characteristic has been derived from the measured transfer characteristic of the device NI-USB-6009 (see figure 8.1).

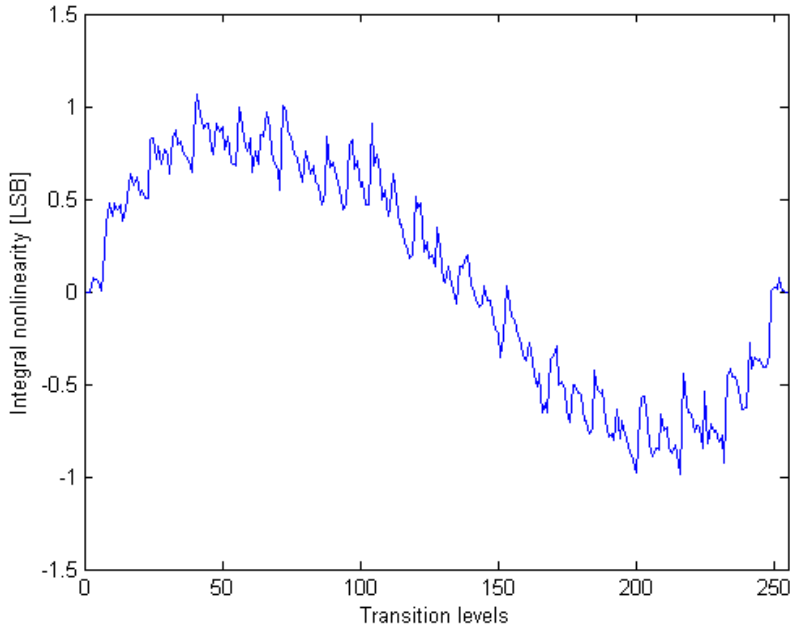

Figure 8.1: *Integral nonlinearity of the simulated ADC*

Amplitude, phase and DC component of the signal was fixed. Each set of simulated measurements contains 20 records, these records are results of simulations with identical signal parameters, and different realization of noise. To observe consistency and asymptotic behavior of estimators, the number of samples increased from 1000 up to 100000. To ensure coherence, in each case 1 total period of the sine wave has been recorded, thus the frequency varied depending on the number of samples: $f/f_s = 1/M$, where M denotes the length of the record.

8.3 Results

8.3.1 Asymptotical behavior

Seven sets of simulated measurement have been processed. The lengths of the records in these sets are 1000, 2000, 5000, 10000, 20000, 50000 and 100000 respectively. Twenty different data vectors have been processed in each set: thus expected values and variances of the estimators have been calculated. As both LS and ML estimators have been computed, behavior of the two different types of estimators can be compared. To create readable illustrations, only three values have been drawn for each estimator: the expected value,

and the $\mu + 3\sigma$ and $\mu - 3\sigma$ boundaries. Figures 8.2, 8.3, 8.4 and 8.5 displays the results achieved: the relative errors of estimators depending on the length of record.

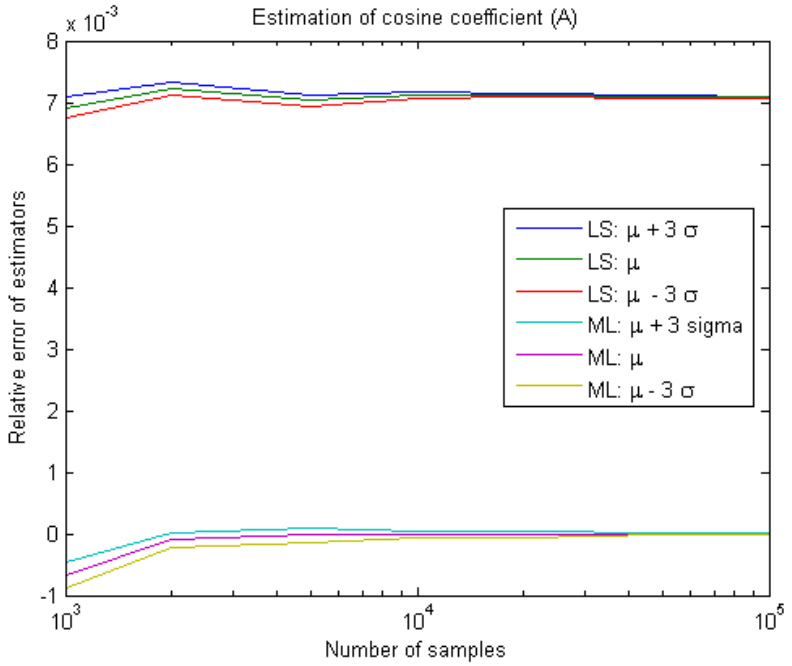

Figure 8.2: *ML and LS estimation of the cosine coefficient*

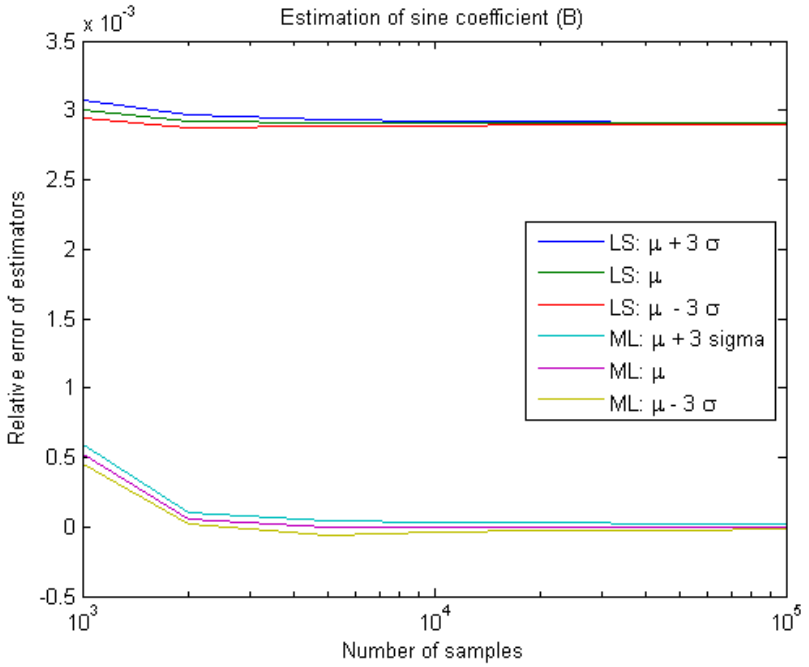

Figure 8.3: *ML and LS estimation of the sine coefficient*

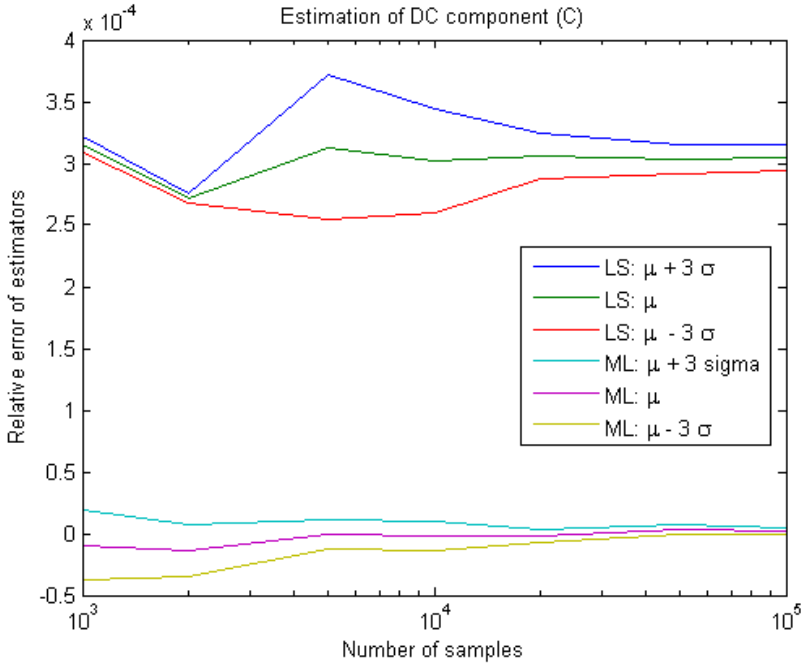

Figure 8.4: *ML and LS estimation of the DC component*

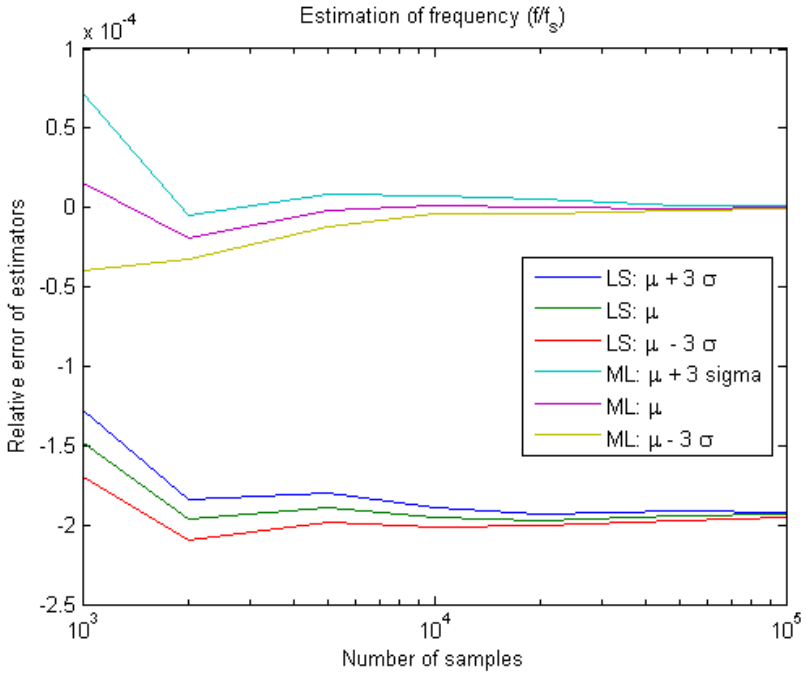

Figure 8.5: *ML and LS estimation of the sine wave frequency*

8.3.2 Variance

The variances of the ML and LS estimators were also compared. To perform this examination, a set of simulated measurements, containing 20 records were used. These records were created using the same signal parameters and the same amount of noise, however the realization of noise was different in each record. Performing ML and LS parameter

estimation for all records provides 20-20 parameter vectors for each method. The covariance of estimators has been approximated using these values. The results are shown below: the diagonals of the covariance matrices are displayed to highlight the most important information: the variances of the estimators.

$$\mathrm{diag}(\hat{\mathbf{C}}_{\mathrm{LS}}) = \begin{bmatrix} 1.1127\text{ e-}07 \\ 3.3468\text{ e-}07 \\ 1.9014\text{ e-}07 \\ 1.1046\text{ e-}09 \end{bmatrix} \tag{8.1}$$

$$\mathrm{diag}(\hat{\mathbf{C}}_{\mathrm{ML}}) = \begin{bmatrix} 4.4242\text{ e-}08 \\ 3.7546\text{ e-}08 \\ 1.0387\text{ e-}08 \\ 6.0340\text{ e-}10 \end{bmatrix} \tag{8.2}$$

where $\hat{\mathbf{C}}_{\mathrm{LS}}$ and $\hat{\mathbf{C}}_{\mathrm{ML}}$ are the estimators of the covariance matrices of LS and ML estimators, respectively. The dimension of the first three parameters is amplitude (voltage or digital code), the fourth parameter is the frequency estimator. Thus the relation between the first three variances and the fourth variance depends on the scaling of frequency. Nevertheless the related values of the ML and LS covariance matrices can be compared, and significantly lower variance of ML estimators can be observed. It is important to remark that variances of the estimators strongly depend on the termination criteria of the optimizing algorithms. Thus this comparison of variances is rather an illustration, and does not prove general statements. Questions concerning the variances of the estimators need further investigation.

8.3.3 Consequences

These results show the difference between LS and ML estimators: as the LS estimator finds the best fitting sine wave in the digital code domain, the nonlinearity of the converter misleads the LS estimators. ML estimators find the most likely sine wave that produces the recorded output: nonlinearity of ADCs is handled in this model. On the other hand ML estimators provide the unbiased estimation of signal parameters with less variance: values of ML estimators are less sensitive to the actual realization of noise on the measurement record.

Chapter 9

Software implementation of ML estimation and other test methods

To perform ADC testing in practice, it is necessary to have software tools to automate the process of data acquisition, signal processing, and evaluation of measurement results. In this section two software tools are presented: one of them is a MATLAB toolbox for offline data processing, the other is a LabVIEW virtual instrument (VI). Both of them are developed by the author: though the ADC Test toolbox is based on the idea of the previous versions of ADC testing software developed by János Márkus, the functionality has been extended very largely, and the program code has been completely re-written (only the four parameter sine wave fit algorithm has been merged from the old toolbox). The LabVIEV VI contains some very important MathScript codes written by Vilmos Pálfi, however data acquisition, dataflow programming, user interface and other MathScript codes have been developed by the author.

9.1 A MATLAB toolbox for ADC testing

The ADC Test toolbox [26] is a set of coherent MATLAB functions using Graphical User Interface (GUI). The goal of this software is to ease the process of ADC testing: using this environment, offline signal processing and measurement evaluation can be performed without any programming or deeper domain knowledge required. This toolbox provides the following main functionalities:

- Assembling measurement descriptors using the data acquired and the circumstances.
- Storing and handling measurement descriptors.
- Processing measurement records and evaluating results in multiple ways, such as
  - Sine wave fitting in least squares sense
  - Histogram testing using sine waves
  - Sine wave fitting using maximum likelihood estimation
  - Frequency domain analysis (FFT test)

9.1.1 The graphical user interface

The main window of the program is presented in figure 9.1. The functionalities appear separately on the screen. Information about the actual measurement descriptor is displayed in the center frame. On the top of the window, the current measurement can be selected from the list of descriptors loaded, using a popup menu. The data handling functionalities appear on the right side of the screen. Data processing possibilities are itemized in the lower frame of the main window.

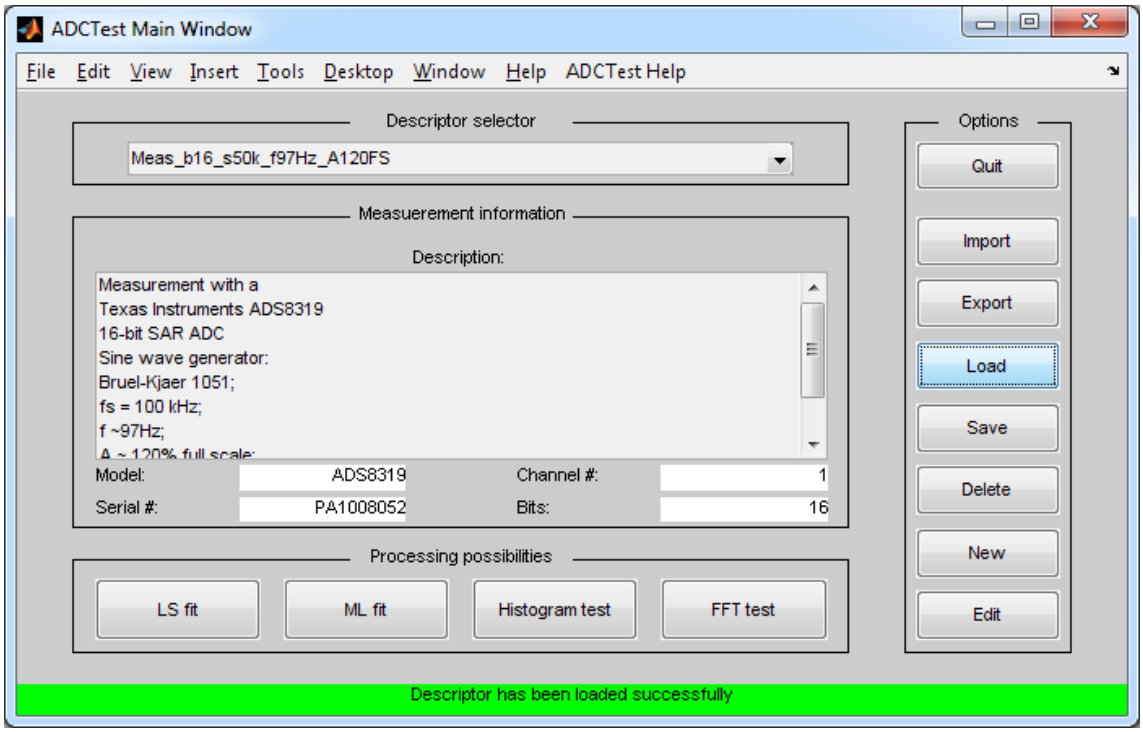

Figure 9.1: *Main window of the user interface*

9.1.2 Data handling functionalities

The toolbox provides the following options to handle measurement descriptors:

- **Import**: Importing entire measurement descriptors from the MATLAB workspace.
- **Export**: Exporting entire measurement descriptors to the MATLAB workspace.
- **Load**: Loading measurement descriptor from a specific XML file.
- **Save**: Saving measurement descriptor to a specific XML file.
- **Delete**: Removing actual measurement descriptor from the memory (does not affect files on disk).
- **New**: Creating new measurement descriptor: it can be assembled using the raw data (imported from the workspace) and the measurement circumstances.
- **Edit**: Editing existing measurement descriptor: more circumstances can be added, or incorrect information can be fixed.

The „New” option also allows the user to perform simulated measurements. This feature can be reached using the „Simulate measurement” pushbutton. Parameters of the simulated measurement can be set such as the parameters of the excitation sine wave, the noise on the analog signal, and the transfer characteristic of the ADC under test. This characteristic can be measured previously, or assembled artificially. In the previous case the vector containing the INL values can be loaded from the MATLAB workspace, in the latter case the INL vector can be assembled from shapes (like sine wave or Hann window) and additive noise (like Gaussian or uniformly distributed).

9.1.3 Data processing possibilities

The main goal of the software tool is to process and evaluate measurements. There are multiple methods to process a measurement for ADC testing, in this toolbox four of them are implemented.

Four parameter sine wave fit in least squares sense

This dynamic test method has been developed to determine the effective number of bits (ENOB), and the signal to noise and distortion ratio (SINAD). The excitation signal is sinusoidal, thus a sine wave shall be fitted to the measurement record. The numerical methods used to perform the fit properly are described in section 2.3. The results are displayed in a dialog box, shown on figure 9.2. The estimated sine wave parameters and the calculated ADC parameters appear numerically, and the fitting residuals are displayed in two ways: statistical properties can be observed via the histogram, and „Mod T” plot shows the location of the residuals in the phase space.

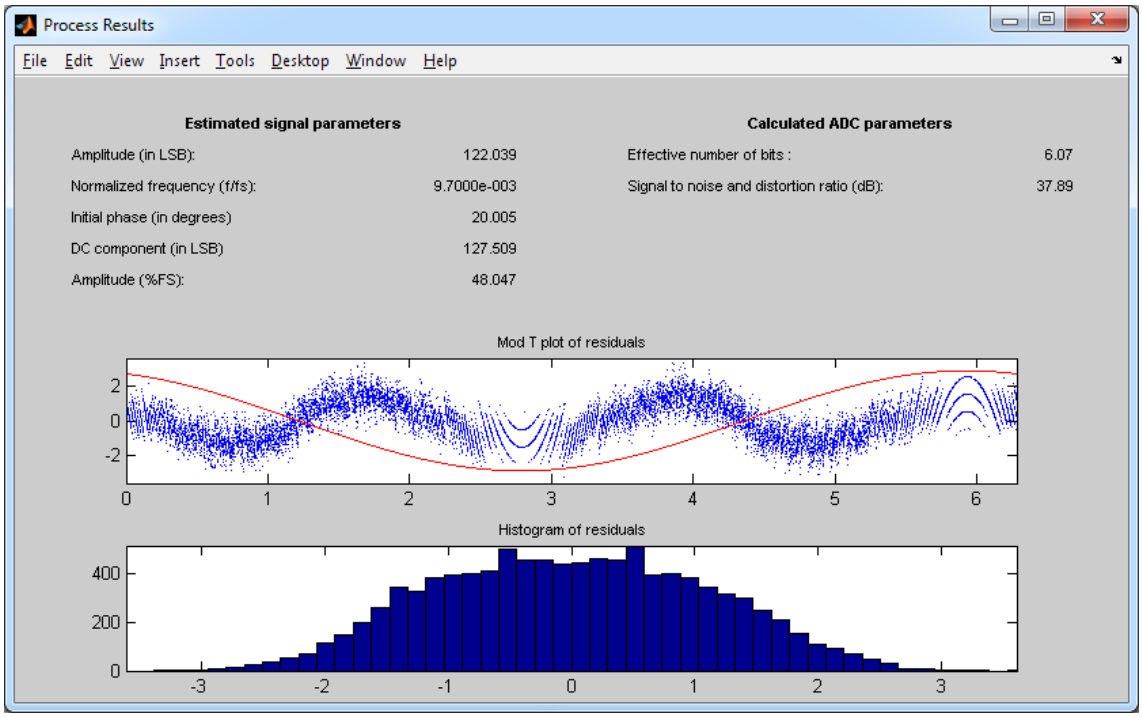

Figure 9.2: *Results window for LS fit*

Histogram test with sinusoidal excitation

Estimation of static transfer characteristic of the ADC is also possible using sine waves. To perform histogram test properly, it is very important to fulfil a few requirements regarding the settings of the excitation signal and the sampling. These requirements are itemized in section 7.1.2. This feature of the toolbox performs two tasks:

- Investigates whether the measurement record is appropriate for histogram testing or not.
- Calculates and displays estimators for integral and differential nonlinearity.

In case of inappropriate measurement records, warning messages appear to notice the user about the problems and the reasons (such as low number of samples, fractional periods, low signal amplitude). Figure 9.3 shows the estimated integral and differential nonlinearity of a 16-bit successive approximation ADC.

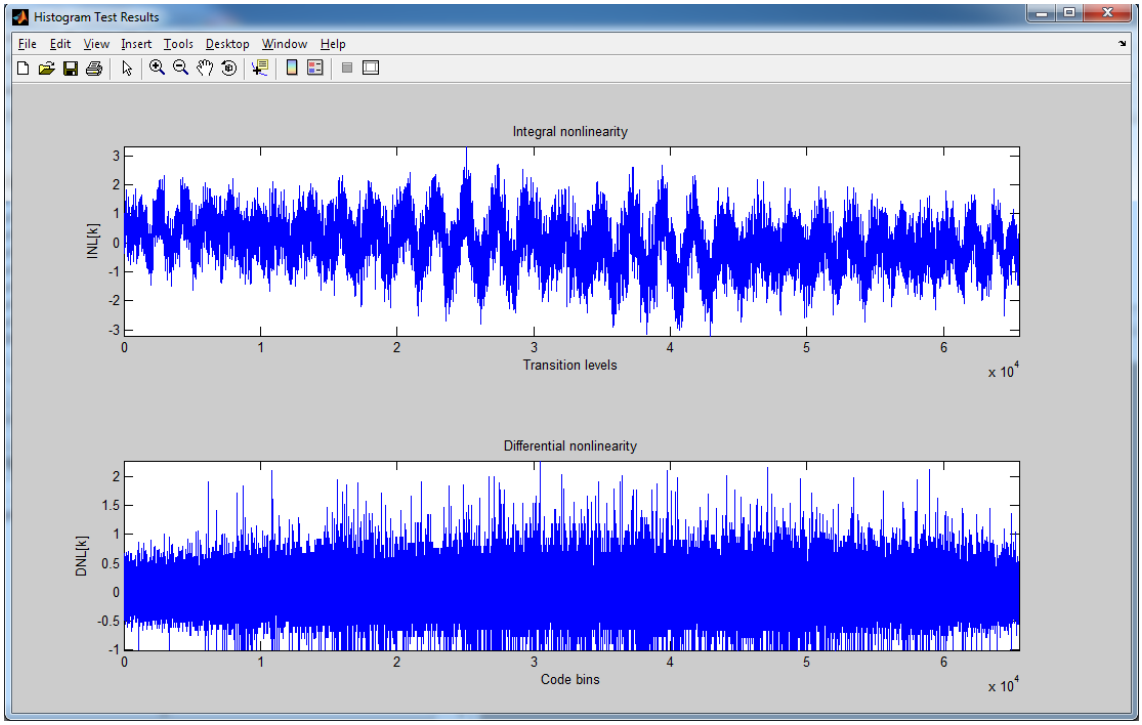

Figure 9.3: *Results window for histogram test*

FFT test

Frequency domain processing of a measurement record is also a useful option. Examining the DFT of the measurement record provides information about the harmonic distortion, the amount of noise and the spurious components. A very important quantity, the *Spurious-free dynamic range, SFDR* of the ADC under test can be calculated using the FFT of the record. In the results window (see figure 9.4), frequencies and magnitudes of the harmonic components appear. The SFDR is calculated relative to the amplitude of the fundamental

sine wave (the carrier), and relative to the full scale (FS) of the converter. The amplitude spectrum of the record is also displayed from DC up to the Nyquist-frequency.

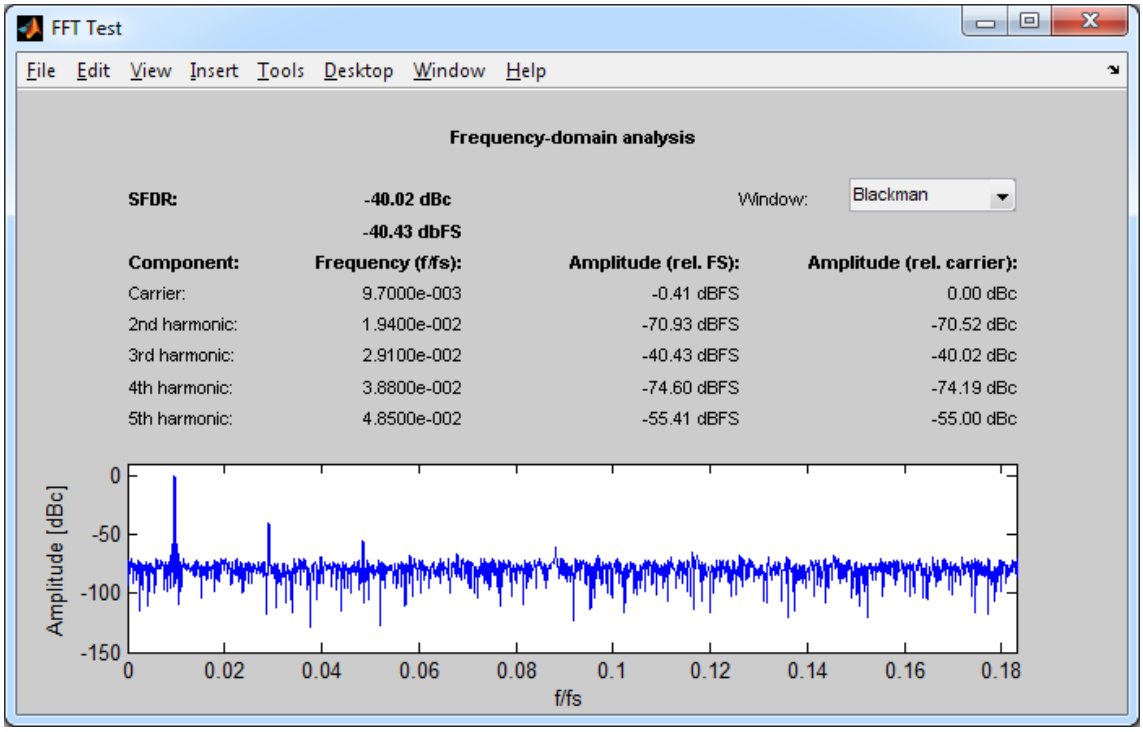

Figure 9.4: *Results window for FFT test*

In frequency domain analysis it is very important to avoid spectral leakage. This toolbox offers three different windowing functions that can be applied in the time domain: Hann, Blackman, and 3-term Blackman-Harris windows are available. Naturally DFT can be performed without windowing, choosing the „None (rect)" option in the „Windowing" popup menu.

Sine wave fitting using maximum likelihood estimation

The most complex task implemented by the toolbox is the maximum likelihood estimation of sine wave parameters. Theoretical fundamentals, practical difficulties and solutions are detailed in the previous chapters. In this section ML estimation will be described only from the user's perspective. To perform estimation, it is required to achieve initial estimators for both the excitation signal parameters and the code transition levels. These can be calculated using four parameter sine wave fit and histogram test. An initial estimator for the noise variance also shall be calculated as it is described in section 6.2. Then extrema of the likelihood function can be reached via numerical optimization. Figure 9.5 shows the framework of ML estimation.

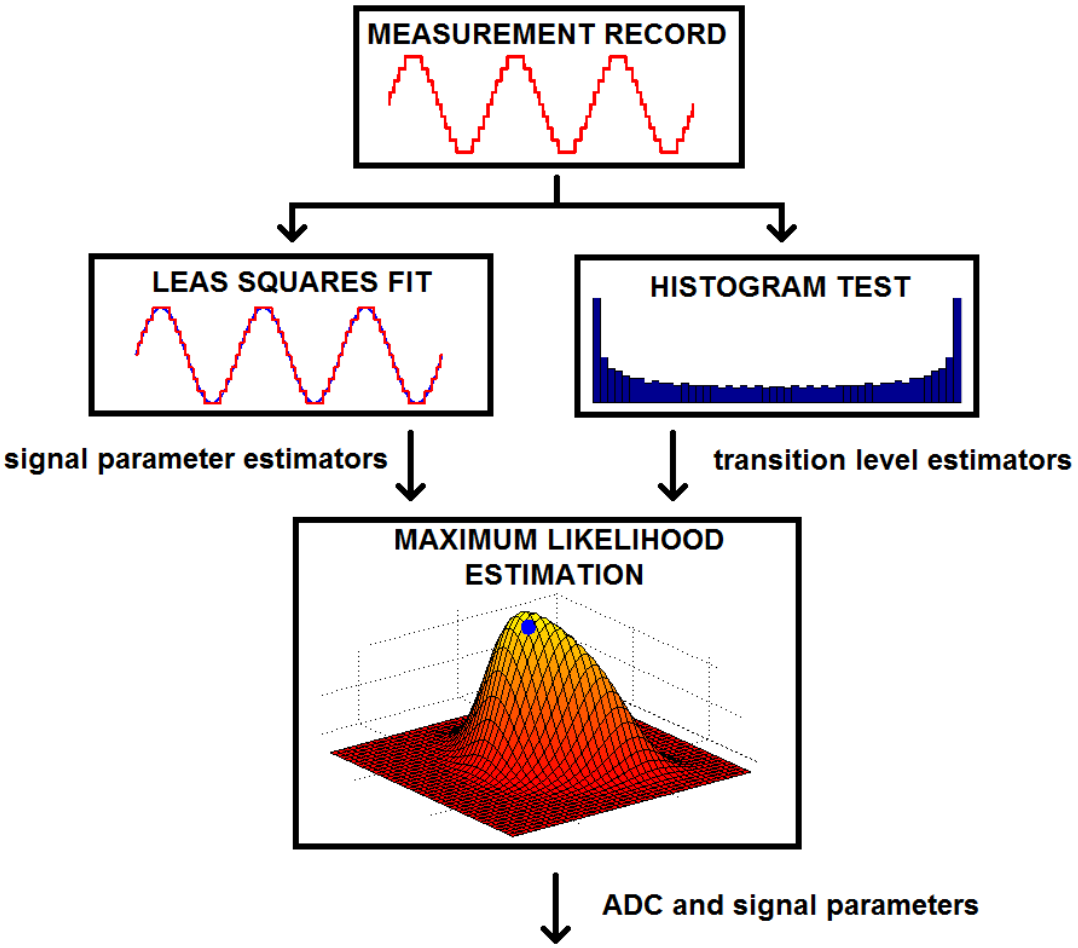

Figure 9.5: *Framework of ML parameter estimation*

Calling „ML fit" from the main window implicitly calls four parameter LS fit and histogram testing routines. Settings of LS fit shall be specified in a dialog box, and histogram test routine displays information (and warnings, if necessary) regarding the appropriateness of the record. The numerical optimization can be followed via the GUI. As shown in figure 9.6, the actual values of the parameter estimators and the cost function are displayed in each iteration cycle. The scaling factor of the *Levenberg-Marquardt* step (λ) changes in each iteration, thus value of λ is also updated continuously. The user can specify the termination criteria for the optimization (such as termination tolerance, maximum number of iterations, or maximum number of cost function evaluations). Iteration can be stopped, paused and resumed using the pushbuttons on the right side of the window.

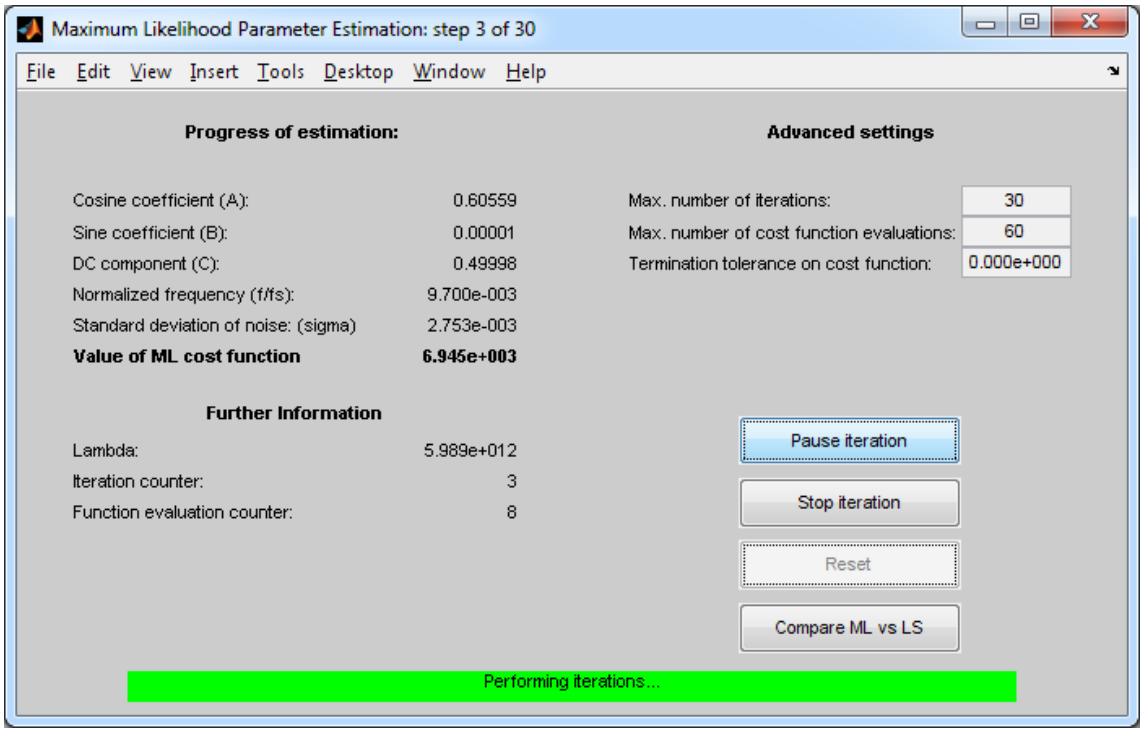

Figure 9.6: *Following iteration cycles of ML estimation*

Extremum of the cost function found by the optimization algorithm provides the ML estimators for the sine wave parameters. This way sine wave fit can be performed using the sine wave estimated in LS sense, and using the sine wave estimated via the maximum likelihood method. ADC parameters (datasheet quantities like ENOB and SINAD) can be calculated using both the ML and LS estimators. This way results of ML and LS fit can be compared easily in a comparison window (see figure 9.7). On the screenshot provided, ML and LS estimation is performed for a nonlinear ADC, thus differences in the fitted sine wave and the calculated device parameters can be observed. Note that the LS fit minimizes the mean squared value of residuals, thus minimizes the power of noise and distortion. This way SINAD and ENOB cannot be higher than their values calculated using LS estimators.

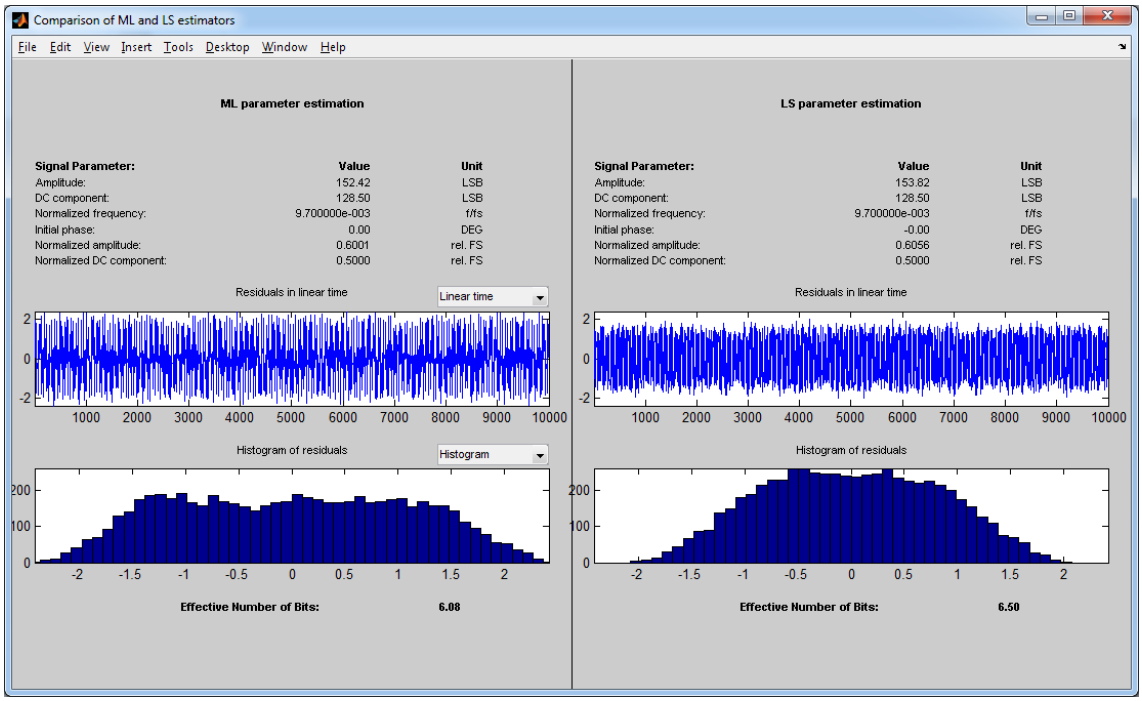

Figure 9.7: *Comparison of results using ML and LS estimators*

9.2 An ADC testing tool for LabVIEW

The LabVIEW environment provides advanced software interfaces to measurement and data acquisition devices. The MathScript module also allows the developer to implement complex signal processing algorithms in a text-based programming language. This way excitation signal generation, data acquisition, and data processing can be integrated into one software component: a virtual instrument (VI). The front panel of the VI (see figure 9.8) provides the user interface. On the front panel

- settings of the signal generator and parameters of the sine wave generated can be specified,
- settings of data acquisition (sample rate, length of record) can be specified,
- data acquisition can be performed and repeated using a pushbutton,
- data processing results are displayed numerically and graphically.

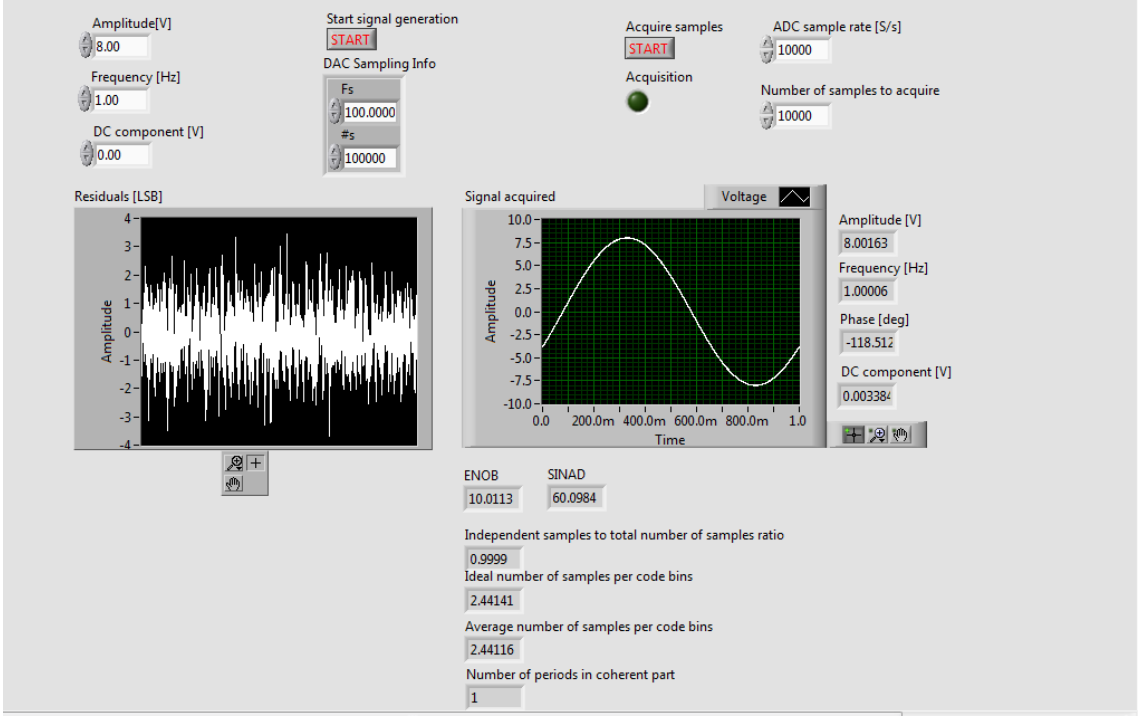

Figure 9.8: *Front panel of the virtual instrument*

To generate analog excitation signal, this VI uses an NI 9263 analog output module containing a 16-bit DAC. The data acquisition device is an NI 9201 analog input module contag a 12-bit ADC. This ADC is the device under test in this project. To write scalable and maintainable code, the complex signal processing algorithms are encapsulated in subVIs. Each subVI performs a general task like sine wave fitting, investigation of coherence, or INL estimation. These elements can be called from multiple different top level VIs depending on the purpose of the application.

Chapter 10

Conclusions

10.1 General consequences

ADC architectures and test methods both have been improved in the last decades. As very sophisticated electrical realizations have been developed, the test methods also became more and more efficient, feasible and robust. This report proposes an advanced mathematical procedure to improve the ADC test results achieved via sine wave fitting. This method is the maximum likelihood estimation of ADC and signal parameters using a measurement record of sinusoidal excitation. It is important to remark that sine wave fitting is only one of the test methods available, however the most meaningful quantities concerning the dynamic behavior of the converter are calculated this way. Values of *signal to noise and distortion ratio* (SINAD) and *effective number of bits* (ENOB) provide clear and unequivocal information regarding the quality of ADC. On the other hand sine wave fitting examines the device under test at a specified frequency: thus properties of the converter can be observed in the frequency domain performing multiple measurements with different sine wave frequencies.

Theoretical background of the maximum likelihood estimation for sine wave fitting has been elaborated and published previously [21]. The results of independent research efforts concern the questions of implementation and practical realization of the ML method. Chapter 6 examines a very important aspect of sine wave fitting in ML sense: the role and effects of noise. Chapter 7 itemizes the main difficulties of the practical realization and proposes solutions for them. Properties of ML estimators are investigated experimentally and compared to the properties of LS estimators (used in standardized methods) in Chapter 8. The novel ML method and other standardized methods have been implemented in two different platforms (MATLAB and LabVIEW) by the author: this software tools are presented in Chapter 9.

In conclusion, maximum likelihood estimation of ADC and signal parameters improve the accuracy of estimators, thus provides more precise datasheet quantities. To perform ML estimation correctly, more efforts (more restrictions on the measurement record, and more computation) are required, however this procedure is also realizable in a simple PC environment (see Chapter 9). Using the ML method is recommended in each cases, it is

worth to take a more efforts to achieve higher accuracy.

10.2 Reflection to the specification

Concerning this M. Sc. thesis, the following tasks were specified by the supervisor:

- **Provide an overview of the properties, advantages and disadvantages of maximum likelihood estimation used for sine wave fitting in ADC testing.** Maximum likelihood estimation has been introduced in Chapters 4 and 5. Disadvantages (challenges of practical realization) have been itemized in Chapter 7. Advantages (properties of ML estimators compared to the ones achieved via LS fit) are detailed in Chapter 8.

- **Develop a MATLAB toolbox that performs ML and LS estimation of sine wave parameters, and provides comparative and illustrative results to examine these techniques. Publish this software tool on the web.** This MATLAB toolbox is described in section 9.1. The software is available on the ADC test project site of the Department of Measurement, and Information Systems (see [26]).

- **Integrate multiple numerical methods into this toolbox, if possible.** The numerical optimization of the ML cost function is described in section 7.2. The core of the algorithm is the evaluation of the cost function and its first and second order partial derivatives. This function has been written and has been embedded into the optimization algorithm, which uses the Levenberg-Marquardt method. The „differential evolution" (DE) algorithm is an other possible option to minimize the cost function. The MATLAB implementation of this numerical recipe is available at [28]. Presently I am investigating the appropriate settings of the DE method to solve this specific problem (minimizing the ML cost function for sine wave fitting). However, this numerical method has not been integrated to the toolbox yet.

- **Prepare the comparison and standardization of test methods implemented in MATLAB and LabVIEW environments in international cooperation.** These test methods have been implemented in MATLAB and LabVIEW environments. An other LabVIEW-based ADC testing software has been developed by the engineers of the Technical University of Kosice ([29]). The need to compare the estimation results achieved using their LabVIEW VIs and our MATLAB toolboxs resulted the following options to exchange data for comparison between the two platforms.

    1. **XML descriptors:** this solution provides a platform-independent and easily readable format to handle measurement descriptors and estimation results (signal parameters). The disadvantage of XML is the ASCII representation: causes large file sizes, slow export and import, and conversion between binary and decimal representation of a number can also be a source of small inaccuracies.

2. **MAT files:** binary MAT file is the default format for MATLAB to store numerical data. MAT files can be read and written faster, store the data more effectively, and contain the information more accurately (in binary representation). MAT files can also be read in LabVIEW using the MathScript module. A VI has been developed and published on the project site to read MAT descriptors in LabVIEW environment.
3. **TDMS files:** TDMS is the preferred binary file format of LabVIEW. As TDMS can store numerical values and related information (such as circumstances of the measurement) in one file, this format can also be used to handle measurement descriptors. TDMS files can be read in MATLAB using a freely available tool([30]), however TDMS import is not integrated to the toolbox yet.

- **Propose a framework to perform comparable analyses using the internet.** Using the data exchange options mentioned above, measurement records can be published, and estimation results can be compared and verified using the toolbox published on the web. Measured and simulated records created by the author are available on the project site in XML and MAT format.

Chapter 11

Outlook

In the previous chapters ML estimation of ADC parameters has been examined from several theoretical and practical viewpoints. However, there are still unanswered questions concerning this area of ADC testing. There are possibilities to improve the model used for ML estimation (these questions can be considered theoretical), and there are other possibilities to develop the practical realization of this method. These topics mentioned above can designate the path of further research.

The scope of investigation can be extended involving these issues:

- Expanding the mathematical model to handle imperfections of sampling: this way sampling jitter can also be estimated.
- Separating the spurious components of the signal according to the source of them: estimating the amount of electronic noise, distortion, and phase noise of the sine wave separately.
- Improving the optimization algorithm of the ML cost function, especially in those cases when the solution of the ML problem is special: the cases of low noise measurements.
- Implementing the ML method in high-performance computing environments, examining the data dependencies and parallelisms of the algorithm. Using general purpose GPUs is very attractive owing to the high-level programming languages such as CUDA or OpenCL.

These questions are closely related to the topic of this paper, however other investigations can be performed in the field of ADC testing based on this report.

Bibliography

[1] Signal swithces and sample-and-hold circuits, *http://mysite.du.edu/ etuttle/electron/elect25.htm*, 2nd December, 2013.

[2] Sample and hold circuit using CA3140 Op-Amp, *http://circuit-diagram.hqew.net/Sample-and-Hold-Circuit-Using-CA3140-Op2dAmp_ 2689.html*, 2nd December, 2013.

[3] Franco Maloberti, *Data Converters*. 2007, Springer. ISBN-10: 0-387-32485-2. ISBN-13: 978-0-387-32485.

[4] University of Cambridge, Faculty of Engineering, Division of Electrical Engineering *http://www.eng.cam.ac.uk/DesignOffice/mdp/electric_ web/Digital/DIGI_ 13.html*, 2nd December, 2013.

[5] All about circuits: architecture of successive approximation ADC, *http://www.allaboutcircuits.com/vol_ 4/chpt_ 13/6.html*, 2nd December, 2013.

[6] Maxim Integrated: Understanding SAR ADCs, *http://www.maximintegrated.com/app-notes/index.mvp/id/1080*, 2nd December, 2013

[7] Analog Devices: Which architecture is right for your application?, *http://www.analog.com/library/analogdialogue/archives/39-06/architecture.html* 5th December, 2013

[8] An introduction to Delta Sigma Converters, *http://www.beis.de/Elektronik/DeltaSigma/DeltaSigma.ht* 2nd December, 2013.

[9] Richard Schreier, Gabor C. Temes, *Understanding Delta-Sigma Data Converters*. 2005, John Wiley & Sons, Inc., Hoboken, New Jersey. ISBN: 0-471-46585-2

[10] Analog Devices, *Analog-Digital Conversion: Testing data converters*, http://www.analog.com/library/analogdialogue/archives/39-06/Chapter%205%20Testing%20Converters%20F.pdf

[11] Hauro Kobayashi et al., *ADC standard and testing in Japanese industry*, Computer Standards and Interfaces 23 (2001) pp. 57-64

[12] E. Balestrieri, P. Daponte, S. Rapuano, *ADC Standard Harmonization: Comparison of Test Methods*, I2MTC 2008 - IEEE International Instrumentation and Measurement Conference, Victoria, Vancouver Island, Canada, May 12-15, 2008

[13] Standard IEEE-1241-2010, *IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters*, 2011

[14] International Standard IEC 60748-4-3, *Interface integrated circuits - Dynamic criteria for analog-to-digital converters*, 2006

[15] International Standard IEC 60748-4, *Semiconductor devices - Integrated circuits - Interface integrated circuits*, 1997

[16] Linus Michaeli, Michal Sakmár, Ján Saliga, *Some errors of analogue signal sources for ADC exponential stimulus histogram test*, Proceedings of the 15th IMEKO TC-4 International Symposium on Novelties in Electrical Measurement and Instrumentation, September 18-22, Iasi, Romania, pp. 51-56

[17] Carni, D.L.; Grimaldi, D.; Michaeli, L.; Saliga, J., *Measurement of the exponential signal distortion*, IEEE Intsrumentation and Measurement Technology Conference (I2MTC), Graz, 13-16th May, 2012. DOI.: 10.1109/I2MTC.2012.6229424

[18] Jan Saliga, Linus Michaeli, Michal Sakmár, Jan Busa, *Processing of bidirectional exponential stimulus in ADC testing*, Measurement, 2010

[19] Jerome Blair, *Histogram Measurement of ADC Nonlinearities Using Sine Waves*, IEEE Transactions on Instrumentation and Measurement, vol. 43, No. 3, June 1994, pp. 373-383.

[20] Schnell, László (Ed.): *Jelek és rendszerek méréstechnikája (Measurement technology of Signals and Systems)*, Műszaki Könyvkiadó, Budapest, 1985. (in Hungarian)

[21] L. Balogh, I. Kollar, A. Sárhegyi, *Maximum likelihood estimation of ADC parameters* I2MTC 2010 - IEEE International Instrumentetion and Measurement Conference, Austin, Texas, USA, May 3-6, 2010

[22] Vilmos Pálfi, István Kollár, *Improving the result of the histogram test using a fast sine fit algorithm*, 19th IMEKO TC 4 Symposium and 17th IWADC Workshop: Advances in Instrumentation and Sensors Interoperability. Barcelona, Spain, 18/07/2013-19/07/2013. Paper 118.

[23] Press, Teukolsky, Vetterling, Flannery, *Numerical Recipes. The Art of Scientific Computing*, Third Edition (2007), 1256 pp.
Cambridge University Press. ISBN-10: 0521880688

[24] Kenneth Levenberg, *A Method for the Solution of Certain Non-Linear Probmems in Least Squares* Quarterly of Applied Mathematics 1944/2: pp. 164-168.

[25] Donald Marquardt, *An Algorithm for Least-Squares Estimation of Nonlinear Parameters* SIAM Journal on Applied Mathematica 1963/11 (2): pp. 431-331.

[26] ADC Test project site: http://www.mit.bme.hu/projects/adctest

[27] Tamás Virosztek, *ADC testing in practice, using maximum likelihood estimation*, „TDK report", *http://mycite.omikk.bme.hu/doc/149632.pdf*

[28] Differential evolution for MATLAB, *http://www.mathworks.com/matlabcentral/fileexchange/18593-differential-evolution*, 12th December, 2013.

[29] Technical University of Kosice - Faculty of Electrical Engineering and Informatics, *http://www.tuke.sk/tuke/faculties-1/fei/top-info?set_language=en&cl=en*, 12th December, 2013.

[30] TDMS reader for MATLAB, *http://www.mathworks.com/matlabcentral/fileexchange/30023-tdms-reader*, 12th December, 2013.