Die approbierte Originalversion dieser Dissertation ist an der Hauptbibliothek der Technischen Universität Wien aufgestellt (http://www.ub.tuwien.ac.at).

The approved original version of this thesis is available at the main library of the Vienna University of Technology (http://www.ub.tuwien.ac.at/englweb/).

DISSERTATION

Digital Pre-distortion of Microwave Power Amplifiers

ausgeführt zum Zwecke der Erlangung des akademischen Grades
eines Doktors der technischen Wissenschaften

eingereicht an der Technischen Universität Wien
Fakultät für Elektrotechnik und Informationstechnik

von

DI Ernst Aschbacher
Herndlgasse 22/34
A-1100 Wien

geboren am 4. September 1970 in Bruneck (IT)
Matrikelnummer: 9326558

Wien, im September 2005

Begutachter:

Univ. Prof. Dr. Markus Rupp
Institut für Nachrichtentechnik und Hochfrequenztechnik
Technische Universität Wien
Österreich

Univ. Prof. Dr. Timo I. Laakso
Signal Processing Laboratory
Helsinki University of Technology
Finland

Abstract

With the advent of spectrally efficient wireless communication systems employing modulation schemes with varying amplitude of the communication signal, linearisation techniques for nonlinear microwave power amplifiers have gained significant interest. The availability of fast and cheap digital processing technology makes digital pre-distortion an attractive candidate as a means for power amplifier linearisation since it promises high power efficiency and flexibility. Digital pre-distortion is further in line with the current efforts towards software defined radio systems, where a principal aim is to substitute costly and inflexible analogue circuitry with cheap and reprogrammable digital circuitry.

Microwave power amplifiers are most efficient in terms of delivered microwave output power vs. supplied power if driven near the saturation point. In this operational mode, the amplifier behaves as a nonlinear device, which introduces undesired distortions in the information bearing microwave signal. These nonlinear distortions degrade the system performance in terms of increased bit error rate and produce disturbance in adjacent channels. A compensation of the nonlinear distortions is therefore of significant importance, not only to keep the system performance high, but also to comply with regulatory specifications regarding the maximum allowed disturbance of adjacent channels. Nonlinear equalisation at the receiver is possible but complicated due to the unknown effects of the channel. Further, this method does not reduce the disturbance in adjacent channels, thus additional analogue filters would have to be placed at the output of the power amplifier. It is therefore natural to reduce the nonlinear distortions at the point where they occur, namely at the transmitter.

Different linearisation methods exist which aim to reduce the nonlinear distortions while keeping the power amplifier in the nonlinear and efficient mode. Traditionally, these techniques employ additional analogue circuitry. Linearisation by digital pre-distortion is a new method which applies digital signal processing techniques for compensating the nonlinear distortions.

Digital pre-distortion splits into three tasks: modelling of the microwave power amplifier, adaptive identification of the model parameters, and development of the pre-distortion filter. These tasks are addressed in this thesis. Further, a prototype system is developed which allows to test the pre-distortion algorithm in real-time using a fixed-point environment.

For the first task, measurements on microwave power amplifier were performed in order to evaluate different models. The difficulty is to find low-complex but at the same time accurate models, which describe not only the nonlinear effects, but account also for the memory effects of the power amplifier.

The adaptive identification of the parameters of two nonlinear models, namely a Volterra model and a Wiener model, is presented thereafter. Gradient-type algorithms are developed and investigated with respect to stability in a deterministic context.

A powerful method for the determination of the pre-distortion filter is presented next. For nonlinear systems it is in general not possible to devise analytic solutions for the pre-inverse which linearises the system for a certain class of input signals. Here, an iterative technique is presented which finds an approximate solution for the pre-inverse.

Based on the developed pre-distortion algorithm, a real-time prototype system is developed. This system proves that the algorithm can be implemented with a limited amount of hardware resources. Further, measurement results show that the algorithm keeps its excellent performance also in an environment with a limited data- and arithmetic accuracy.

Kurzfassung

Mit der Einführung von spektral effizenten drahtlosen Kommunikationsystemen, die Modulationsformate einsetzen, die auch die Amplitude des Nachrichtensignals verändern, gewannen Linearisierungsverfahren für nichtlineare Mikrowellen-Leistungsverstärker immer mehr an Interesse. Die Verfügbarkeit von schneller und billiger digitaler Signalverarbeitungstechnologie macht die digitale Vorverzerrung als Methode zur Linearisierung von Leistungsverstärkern sehr attraktiv, da sie hohe Leistungseffizienz und hohe Flexibilität verspricht. Digitale Vorverzerrung steht weiters im Einklang mit den gegenwärtigen Bestrebungen eine möglichst Software-definierte Funkübertragung zu schaffen. Ein wichtiges Ziel hierbei ist, kostspielige und unflexible analoge Schaltkreise auf das unbedingt nötige Maß zu reduzieren und durch billige und re-programmierbare digitale Technologie zu ersetzen.

Mikrowellen-Leistungsverstärker sind am effizientesten wenn sie in der Sättigung betrieben werden. Effizienz heißt hier, daß möglichst viel zugeführte Leistung in abgegebene Mikrowellenleistung umgesetzt wird. In dieser Betriebsweise ist der Verstärker nichlinear, was zu Verzerrungen im informationstragenden Mikrowellensignal führt. Diese ungewollten Verzerrungen führen zu erhöhten Bitfehlerraten und verschlechtern somit die Güte des Funksystems. Weiters werden Störungen in benachbarte Frequenzbänder emittiert. Eine Kompensation dieser Signalverzerrungen erhöht demnach nicht nur die Güte des Funksystems, sondern macht das System konform mit Regulationsspezifikationen. Eine nichtlineare Entzerrung des Signals am Empfänger ist möglich aber kompliziert wegen der unbekannten Effekte des Funkkanals. Diese Methode reduziert weiters nicht die Störungen in benachbarten Frequenzbändern. Zusätzliche analoge Filter müssen am Sender eingesetzt werden. Es ist daher natürlich, die Verzerrungen an der Stelle ihres Auftretens, nämlich am Sender, zu kompensieren.

Verschiedene analoge Linearisierungsverfahren existieren. Sie versuchen die Verzerrungen bei gleichzeitigem nichtlinearen Betrieb des Verstärkers zu kompensieren. Digitale Vorverzerrung ist eine neue Methode, die digitale Technologie zur Entzerrung des Verstärkes einsetzt.

Digitale Vorverzerrung kann in drei Teile aufgeteilt werden: Modellierung des Leistungsverstärkers, adaptive Identifikation der Modellparameter und Entwicklung des Vorverzerrungsfilters. Alle drei Aufgaben werden in dieser Dissertation behandelt. Weiters wird ein Prototyp-System entwickelt, das es gestattet, den Vorverzerrungsalgorithmus in Echtzeit in einer Fixpunkt Umgebung zu testen.

Für die erste Aufgabe wurden Messungen mit verschiedenen Leistungsverstärkern ausgeführt, um verschiedene Modelle gegeneinander zu evaluieren. Die Schwierigkeit besteht dabei darin, möglichst genaue und gering komplexe Modelle zu finden, die nicht nur das nichtlineare Verhalten des Verstärkers sehr genau beschreiben, sondern auch Speichereffekte berücksichtigen.

Die adaptive Identifikation der Modellparameter von zwei Modellen, eines Volterramodells und eines Wienermodells, wird danach behandelt. Gradientenverfahren werden entwickelt und auf Stabilität in einem deterministischen Kontext untersucht.

In weiterer Folge wird eine Methode entwickelt um das Vorverzerrungsfilter für ein gegebenes Verstärkermodell zu bestimmen. Analytische Lösungen sind im Allgemeinen nicht verfügbar. Hier wird eine iterative Methode präsentiert, die eine Näherungslösung für das ideale Filter findet.

Basierend auf diesem iterativen Algorithmus wird ein echtzeitfähiger Prototyp entwickelt. Dieses System zeigt, daß der entwickelte Vorverzerrungsalgorithmus mit einem beschränken Aufwand realisiert werden kann. Messungen belegen, daß der Algorithmus seine exzellente Güte auch in einer Umgebung mit beschränkter Zahlen- und Arithmetikgenauigkeit beibehält.

Acknowledgement

Several people contributed in various ways to this thesis. I am deeply indebted to and would like to express my gratitude to:

- *Prof. Markus Rupp* for his continuous support and guidance. His lucid comments and critical questions improved the scientific quality of this thesis significantly.
- *Prof. Timo I. Laakso* for assuming the work as a referee.
- *Mei Yen Cheong* – intensive discussions with her provided a basis for a deeper understanding.
- My diploma students *Mathias Steinmair* and *Peter Brunmayr* for their questions and work.
- *Holger Arthaber* and *Michael Gadringer* who kindly assisted me with microwave hardware and measurement equipment.
- *Anding Zhu* and *Prof. Tom Brazil* for hosting me at the University College Dublin and for fruitful discussions.

Finally, and most importantly, I thank my family for their support and company.

Contents

1 Introduction **1**
- 1.1 Outline of the Thesis and Contributions . . . 3
- 1.2 Power Amplifier Linearisation: From Analogue Linearisation to Digital Pre-distortion . . . 4
  - 1.2.1 Motivation . . . 4
  - 1.2.2 Analogue Techniques . . . 5
    - Power Back-off . . . 5
    - Feedforward Linearisation . . . 5
    - Cartesian-Loop . . . 6
  - 1.2.3 Digital Pre-distortion . . . 7
    - Digital Pre-distortion: Building Blocks . . . 8
    - Digital Pre-distortion – Brief Literature Review . . . 9

2 Power Amplifier Modelling **11**
- 2.1 The Volterra Series . . . 12
  - 2.1.1 Complex Baseband Volterra Series . . . 13
  - 2.1.2 Frequency Domain Representation of a Volterra Series . . . 15
  - 2.1.3 Discrete-time Volterra Series . . . 16
  - 2.1.4 Series Representation of a Static Non-linearity . . . 18
  - 2.1.5 Parameter Estimation for the Volterra Model . . . 19
- 2.2 The Wiener Model . . . 20
  - 2.2.1 Parameter Estimation for the Wiener Model . . . 21
- 2.3 The Hammerstein Model . . . 22
  - 2.3.1 Parameter Estimation for the Hammerstein Model . . . 23
- 2.4 The Saleh Model . . . 24
  - 2.4.1 Parameter Estimation for the Saleh Model . . . 25
- 2.5 Model-Structure Selection and Model Validation . . . 25
  - 2.5.1 Model-Structure Selection . . . 26
  - 2.5.2 Model Validation . . . 26
- 2.6 Modelling Measured and Simulated Power Amplifiers . . . 27
  - 2.6.1 Black-Box Modelling of Three Microwave Power Amplifiers . . . 28
    - Modelling a Three-Stage High-Power LDMOS EDGE Amplifier . . . 29
    - Modelling a Single-Stage Medium-Power Amplifier . . . 32
    - Modelling a Simulated WCDMA Power Amplifier . . . 34
- 2.7 Discussion . . . 35

3 Adaptive Identification 37

3.1 The Stochastic-Gradient Algorithm . . . 38
  3.1.1 Stochastic-Gradient Identification for Linear-in-Parameter Models . . . 38
    Convergence Analysis – Error-Vector in the Mean . . . 40
    Deterministic Robustness Analysis – Local Passivity . . . 41
    Deterministic Robustness Analysis – Global Passivity . . . 42
    Feedback Structure – Local and Global Passivity . . . 42
    Example – Learning Behaviour of an Adaptive Volterra Filter . . . 43
  3.1.2 Stochastic-Gradient Identification of a Wiener System . . . 45
    First Step: Parameter Estimation for Linear Filter . . . 46
      Local Passivity Relations . . . 48
      Feedback Structure – Local Passivity . . . 49
      Global Passivity Relations . . . 50
      Example 1: Identification of the Linear Part of a Wiener System – Local Stability . . . 51
      Example 2: Identification of the Linear Part of a Wiener System – Global Stability . . . 52
    Second Step: Parameter Estimation for Static Nonlinear Filter . . . 55
      Example 1 continued: Identification of the Nonlinear Part of a Wiener System . . . 57
    The Complete Algorithm . . . 58
3.2 Discussion . . . 58

4 Linearisation by Pre-distortion 61

4.1 Formulation of the Linearisation Problem as an Optimisation Problem . . . 63
4.2 The Fixed-Point Approach . . . 64
4.3 Linearisation using the Secant Method . . . 66
4.4 Applying the Linearisation Methods . . . 69
  4.4.1 Comparison of the Convergence Rate . . . 70
  4.4.2 System-Model Mismatches . . . 73
    Influence of Measurement Noise . . . 73
    Neglecting Memory Effects . . . 74
    Underestimating the Nonlinear Order . . . 75
  4.4.3 Linearising a High-Power LDMOS EDGE Amplifier – Measurement Results . . . 76
    Modelling . . . 78
    Linearisation . . . 79
4.5 Discussion . . . 82

5 Prototype Implementation 85

5.1 System Concept . . . 86
5.2 Signal Processing Hardware – Description . . . 86
  5.2.1 Assembled Digital System . . . 90
5.3 Radio-Frequency Part – Description . . . 92
5.4 Implementation Details . . . 94
  5.4.1 Pre-distortion Filter Implementation . . . 94
    Division . . . 95
  5.4.2 Model-parameter Estimation . . . 97

5.5 Measurements and Performance Results 97
5.5.1 Measurement Results: Floating-point vs. Fixed-point 97
5.5.2 Measurement Results: Real-Time Implementation 98
5.5.3 Used FPGA Resources 102
5.6 Discussion 102

6 Conclusions **103**

A Appendix: Adaptive Identification **105**
A.1 Derivation of (3.40) 105
A.2 Derivation of (3.67) and (3.68) 106
A.3 Derivation of (3.73) 106
A.4 Derivation of (3.86) 107

B Appendix: Linearisation by Pre-distortion **109**
B.1 The Contraction Mapping Theorem 109

C Appendix: Prototype Implementation **111**
C.1 Convergence of the Newton-Raphson Method Applied for Division 111
C.2 Error-Analysis of the Newton-Raphson Method Applied for Division 112
C.3 The Prototyping-Hardware 113

D Appendix: Abbreviations and Symbols **117**
D.1 List of Abbreviations 117
D.2 List of Mathematical Symbols 118

Bibliography **119**

Chapter 1

Introduction

Amplification of information bearing signals is an integral part of every wireless transmitter. The aim is to boost the signal to a sufficient power level for transmission in order to supply the receiver with a sufficiently high level of signal power. Despite of disturbances and signal distortions, the receiver has the task to retrieve the information from the received signal.

In wireless communication systems such as mobile communication systems (e.g., GSM[1], UMTS[2], WLAN[3]) and satellite communication systems (e.g., radio and television broadcast satellites) an essential constraint implies that communication in other frequency bands must not be disturbed excessively. Further, efficient conversion of supplied power into radiated signal power is a key requirement, especially in satellite communication systems where not only power supply is limited, but also heat development becomes a serious technical problem. In mobile communications, efficiency is of particular importance in mobile phones – the power amplifier still consumes the largest amount of energy despite an excessive and increasingly complex digital circuitry. Efficiency in the base station stands for lower operating costs due to a reduced energy supply and smaller cooling units. A common measure for power amplifier efficiency is the power added efficiency (PAE), defined as

$$\mathrm{PAE} \triangleq \frac{P_{\mathrm{RF,out}} - P_{\mathrm{RF,in}}}{P_{\mathrm{DC}}} ,$$

where $P_{\mathrm{RF,out}}$ indicates the output power, $P_{\mathrm{RF,in}}$ is the input power of the power amplifier (at radio frequency), and P_{DC} is the supplied power. Typical efficiencies achieved today in mobile communication systems are 20 % for a UMTS base station amplifier, and 40 % for a UMTS mobile unit.

The two constraints, distortion-free amplification and efficient amplification complicate the simple-sounding task of boosting a signal to a high power level. Ideal distortion-free amplification and efficiency tend to be mutually exclusive. Improvements in efficiency are achieved at the expense of distortions, and vice versa.

---

[1] Global System for Mobile communications. It is the second generation (2G) of wireless mobile communication systems employing digital modulation technology.

[2] Universal Mobile Telecommunications System. This is the European entrant for third generation (3G) mobile communication systems and subsumed in the IMT-2000 family as the WCDMA (Wideband Code Division Multiple Access) technology.

[3] Wireless Local Area Network, a short range radio network normally deployed in traffic hotspots such as airport lounges, hotels and restaurants. WLAN enables suitably equipped users to have wireless access to a fixed network, providing high speed access (up to 54 Mbit/s download) to distant servers. The key WLAN technologies is the IEEE 802.11 family.

Signal distortion in power amplifiers occurs due to two mechanisms: **non-linearity** and **dispersion**. Efficiency requirements push the power amplifier into the nonlinear operational regime, whereas the dispersion effects have their origin in internal memory effects of the active device (i.e., the transistor) and in non-ideal matching networks exhibiting a frequency dependent behaviour. In order to observe dispersion in a well designed power amplifier, the signal bandwidth has to be large, i.e., it must cover variations in the frequency response of the power amplifier (if assumed linear). The signal bandwidth of communication systems increase with every generation: from 200 kHz in GSM (second generation) to approximately 5 MHz (for one carrier) in UMTS, the third generation mobile communication system. This is a 25-fold bandwidth increase. In the IEEE 802.11n WLAN standard even a transmission bandwidth of 40 MHz is specified. If more communication channels have to be amplified (e.g., in a multi-carrier base station) the requirements on linearity and dispersion increase further. Amplification of more than one carrier using only one multi-carrier power amplifier has several advantages over the more traditional approach of amplifying each carrier separately and combining the amplified signals before the common antenna. Before combination, each amplified signal has to pass an isolator and a filter, which must have a high quality factor due to its small relative bandwidth. These narrowband filters are space-consuming, lossy, and difficult to retune, which is necessary to accommodate different choices of transmitted carrier frequencies, making it difficult for system operators to implement a dynamic channel allocation. A second and very important reason for implementing multi-carrier power amplifiers is the achieved forward-compatibility with future systems. The system operator has the possibility to change the modulation scheme without replacing any amplifier. Such multi-carrier power amplifiers must be designed to permit amplification of different modulation schemes at the same time.

Linearisation methods can increase efficiency indirectly by admitting the usage of efficient nonlinear power amplifiers. Depending on the linearisation scheme, linear operation is achieved via an appropriate modification of the input or output signal of the power amplifier. Further, the linearisation scheme must also be capable of compensating the signal dispersion. It is therefore not simply a linearisation scheme – it is a (nonlinear) equalisation task that has to be performed. In order to increase effectively the efficiency of the whole system – linearisation subsystem and power amplifier – it is essential to design low-complex and power-efficient linearisation schemes, maintaining at the same time a certain degree of flexibility and compatibility. Different linearisation schemes, working entirely in the analogue domain or applying a digital approach using digital signal processing techniques, have been proposed and are already in use in communication systems.

A promising candidate for microwave power amplifier linearisation is digital pre-distortion. The method works entirely in the digital domain which is attractive since the hardware implementation employs standard and cost-efficient components. A high degree of flexibility is guaranteed if reprogrammable hardware, such as DSPs[4] and/or FPGAs[5] is used, which is not only desirable in a first prototype development, but also in a final product in order to maintain the possibility to efficiently adjust for later system changes. The costly analogue part in a transmitter is minimally augmented applying this technique. The trend of moving the flexible digital part of a transmitter as close to the antenna while reducing the analogue front-end to its necessary minimum naturally leads to digital pre-distortion as a means for linearisation.

---

[4] DSP stands for Digital Signal Processor. It is a special type of processor micro-architecture, designed for performing mathematical operations involved in digital signal processing.

[5] FPGA stands for Field Programmable Gate Array, a type of chip where the logic function can be defined by the user. Due to this capability and the large number of gates it is especially suited for prototyping of complex communication systems.

1.1 Outline of the Thesis and Contributions

In the following the organisation of the thesis including the contributions of the author is presented.

Chapter 1 Following the motivation for power amplifier linearisation, an overview of commonly used analogue linearisation schemes [1, 2] is presented in Section 1.2. The concept of digital pre-distortion, originally proposed in [3], as an efficient and flexible digital linearisation scheme is introduced.

Chapter 2 Models for power amplifiers are discussed in this chapter. The aim is to have an accurate and relatively simple model, which not only describes the non-linearity of the power amplifier, but also the dynamic effects (memory effects). The classical Volterra series is introduced [4, 5, 6, 7, 8] as the most general model for describing nonlinear dynamic systems. Simpler models, especially Wiener- and Hammerstein models, a specialisation of the Volterra series, are introduced next. Static models, such as the often used Saleh-model [9], as well as series expansion are introduced for comparison reasons. A complex baseband description of the models is derived. Comparisons of performed measurements on real microwave power amplifiers with the presented models are evaluated [10, 11, 12]. A discussion of the shortcomings and difficulties associated with the models concludes the chapter.

Chapter 3 Adaptive identification of the power amplifier is necessary in order to identify the model parameters and to track changes of the system behaviour over time. Emphasis is on low complex and robust gradient-type identification schemes, which are predominantly used in practice. An introduction is devoted to the analysis methods for adaptive algorithms [13, 14]. A deterministic robustness analysis of an adaptive Volterra filter is then performed. The formulation of a two-step adaptive gradient-type algorithm for the identification of the parameters of a Wiener system follows. A robustness analysis for the derived algorithm is conducted [15, 16].

Chapter 4 The problem of equalisation of a nonlinear dynamic system is addressed. Since most often analytic solutions describing the inverse of the nonlinear system are not known, different approaches are proposed in the literature which solve the problem in an approximate way [17, 18]. A new iterative method is developed here, which is simple and converges very fast to a good solution [19, 12]. By means of measurements on a high-power microwave amplifier, the developed linearisation method is tested and the concept is proven to work in an experimental setup on a real physical system [12].

Chapter 5 A prototype system based on an FPGA implementation of the proposed linearisation method was developed [20]. The implementation allows to test the developed linearisation algorithm [12, 21] in real-time using various power amplifier models. It is a very flexible environment for the evaluation of digital pre-distortion for different power amplifiers requiring different models. Measurement results prove the functionality of the implementation and prove that a real-time fixed-point implementation of the proposed algorithm is indeed feasible.

1.2 Power Amplifier Linearisation: From Analogue Linearisation to Digital Pre-distortion

In this chapter an overview of power amplifier linearisation techniques is presented. Starting with a motivation for linear microwave amplification, traditional analogue techniques, such as feedforward and Cartesian-loop linearisation schemes are briefly presented. The concept of digital pre-distortion is introduced. The chapter concludes with a literature review.

1.2.1 Motivation

Highly linear transmitters are forced mainly by:

1. regulatory and

2. system requirements.

Regulatory requirements define severe limits of out-of-band radiation in order to not disturb neighbouring channels excessively. The transmitted signals have to adhere to specific spectral masks, see, e.g., [22], which defines the base station minimum radio-frequency (RF) requirements of the FDD[6] mode of WCDMA[7].

System requirements on the linearity of power amplifiers are especially important in the case of multi-channel power amplifiers. Multi-channel power amplifiers must meet stringent linearity requirements in order to keep the cross-modulation between the channels at a low level, at the same time the system is required to meet these linearity requirements over a large bandwidth of tens of MHz. Such broadband signals resolve the dynamic effects of the power amplifier, a frequency-dependent amplification being the result. These are challenging requirements that prevent often the usage of a multi-channel power amplifier and favour the usage of a channelised approach, which has the advantage that each single amplifier has to meet only moderate linearity requirements over a relatively small bandwidth, thus appear as static devices.

Linearisation methods aim to ameliorate the situation by adding additional circuitry for reducing the nonlinear distortions of the power amplifier. Traditionally, analogue linearisation schemes have been used, the most common, especially in base stations, being the feedforward scheme. The demand for higher flexibility and lower cost with similar performance as analogue linearisation schemes leads to the concept of digital pre-distortion. Signal processing techniques, which can be efficiently implemented using digital hardware such as Digital Signal Processors (DSPs) and/or Field Programmable Gate Arrays (FPGAs), are used to control an analogue RF-system. The advantage is that a high degree of flexibility is maintained due to the inherent flexibility of the digital hardware which allows for changes at run-time of the system. This is in line with the current trend to Software Defined Radio (SDR) [23], where the ultimate goal is to define highly reconfigurable radios which can accommodate a variety of standards and transmission/receive modes, controlled entirely by software. This is only possible if the inflexible and costly analogue circuitry is reduced to a minimum by replacing as much as possible by reprogrammable digital hardware.

---

[6] FDD stands for Frequency Division Duplexing, an application of frequency-division multiple access, used to separate transmit and receive signals in the frequency domain.

[7] WCDMA stands for Wideband Code Division Multiple Access, the technology used by UMTS.

1.2.2 Analogue Techniques

Three analogue linearisation techniques, power back-off, feedforward linearisation, and Cartesian-loop linearisation are briefly presented.

Power Back-off

The conventional and simplest approach for achieving highly linear amplification is to use a class A [1] power amplifier, being inherently inefficient with small input power levels, and to feed it with an input power far below its (efficient) capabilities. This results in very low power-efficient and oversized amplifier systems, making this approach very inefficient for most applications.

Feedforward Linearisation

Feedforward linearisation is a very old method, dating back to 1928 [24, 25]. Nevertheless, feedforward linearisation is used extensively in nowadays base stations [2, 1], since it is a mature technology and provides good linearisation performance.

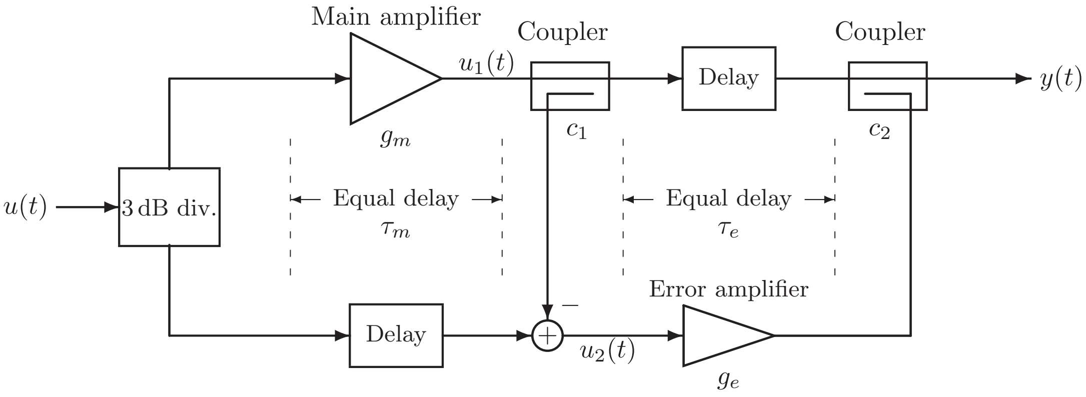


Figure 1.1: Feedforward linearisation scheme

Fig. 1.1 illustrates the feedforward linearisation scheme. It is inherently stable since no feedback path exists. The nonlinear distortions are corrected at the output, in contrast to feedback systems. The price for the potentially high performance and inherent stability is the poor efficiency of the overall scheme. A highly linear and thus power-inefficient error-amplifier is needed, as well as highly precise analogue RF-components, especially delay-lines. These analogue components are required to maintain the accuracy over loading, time, and temperature [26, 27].

A very simple calculation of the output signal $y(t)$ clarifies the operation of the feedforward scheme. The signal $u_1(t)$ is

$$u_1(t) = \frac{1}{2} g_m u(t - \tau_m) + v_{\mathrm{NL}}(t) \; , \tag{1.1}$$

where $u(t)$ is the microwave input signal, τ_m is the delay of the main amplifier, and $v_{\mathrm{NL}}(t)$ models the introduced nonlinear distortion of the main amplifier. The attenuation of the 3 dB splitter is taken into account via the factor $\frac{1}{2}$. The signal $u_2(t)$ is

$$u_2(t) = \frac{1}{2}\big(1 - c_1 g_m\big)u(t - \tau_m) - c_1 v_{\rm NL}(t) \tag{1.2}$$

with the coupling-factor c_1. The output signal $y(t)$ is

$$y(t) = u_1(t - \tau_e) + c_2 g_e u_2(t - \tau_e)\ , \tag{1.3}$$

where τ_e is the delay of the error-amplifier, g_2 is the gain of the error-amplifier, and c_2 is the coupling-factor of the coupler at the output. After a simple calculation the output signal

$$y(t) = \frac{1}{2}\big(g_m + c_2 g_e - c_1 c_2 g_m g_e\big)u(t - \tau_m - \tau_e) + \big(1 - c_1 c_2 g_e\big)v_{\rm NL}(t - \tau_2) \tag{1.4}$$

follows. If $g_e = \frac{1}{c_1 c_2}$ the distortion is completely removed and the output signal is simply

$$y(t) = \frac{1}{2c_1}u(t - \tau_m - \tau_e)\ . \tag{1.5}$$

The coupling-factor c_1 determines the overall gain. An obvious choice is $c_1 = \frac{1}{g_m}$, which results in

$$u_2(t) = -\frac{1}{g_m}v_{\rm NL}(t) \tag{1.6}$$

and

$$y(t) = \frac{g_m}{2}u(t - \tau_m - \tau_e)\ . \tag{1.7}$$

In this analysis the delays of the delay lines are assumed to be perfectly matched to the introduced delays of the amplifiers. This perfect match is in practice difficult to maintain over varying operating conditions, reducing the performance significantly [27].

Cartesian-Loop

The Cartesian-loop linearisation scheme [28, 29, 30] is a feedback scheme, thus able to track system changes. Fig. 1.2 shows the scheme. The main components are the feedback-path and the error-amplifiers. The arrangement will force the output signals $I_{\rm out}(t), Q_{\rm out}(t)$ to track the input signals $I(t), Q(t)$. The performance of the scheme depends on the gain and bandwidth of the video-circuitry, and on the linearity of the I-Q de-modulator [31].

With the availability of low-cost and precise quadrature modulators and de-modulators, the system appears as a simple and attractive architecture. Over a limited range of bandwidths (tens of kHz), linearity improvements up to 45 dBc have been achieved [1, 30].

A simple analysis reveals the operation principle of this scheme. The input signal $u(t) = I(t) + jQ(t)$ is introduced. Further, $v_{\rm NL}(t)$ denotes the additive nonlinear distortions produced by the power amplifier with gain g_m. It can easily be shown that the output signal is

$$y(t) = \frac{g_m g_e}{1 + g_L}u(t) + \frac{1}{1 + g_L}v_{\rm NL}(t) \tag{1.8}$$

with the loop-gain

$$g_L = g_m g_e c\alpha\ , \tag{1.9}$$

composed of the coupling-factor c, the attenuation α, the gain of the error-amplifiers g_e, and the gain of the main power amplifier g_m. For a large loop-gain, (1.8) reduces to

$$y(t) \approx \frac{1}{c\alpha} u(t) + v_{\mathrm{NL}}(t) \frac{1}{g_L} \; . \tag{1.10}$$

If $\frac{1}{c\alpha} = g_m$, this simplifies further to

$$y(t) \approx g_m u(t) + v_{\mathrm{NL}}(t) \frac{1}{g_e} \; . \tag{1.11}$$

The suppression of the nonlinear disturbance depends in this case on the gain of the error-amplifiers g_e.

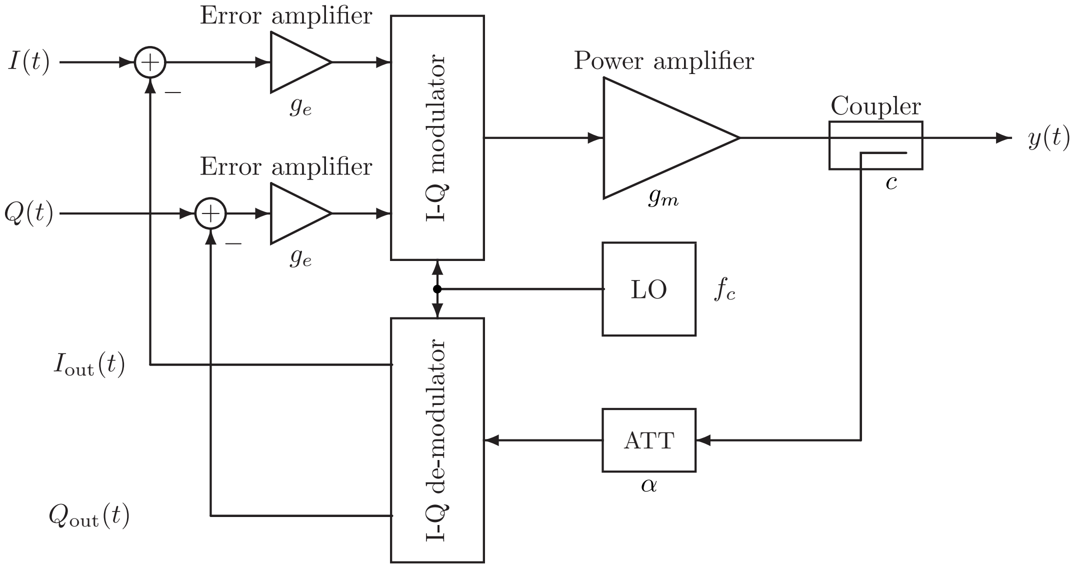


Figure 1.2: Cartesian-loop linearisation scheme

1.2.3 Digital Pre-distortion

The Cartesian-loop scheme provides a motivation for digital pre-distortion. The idea is to build a correction entity which compensates the non-linearity of the power amplifier in digital baseband. The advantages by using digital techniques are a reduced analogue circuitry, excellent reproducibility and precision, as well as the high flexibility.

Fig. 1.3 shows a digital pre-distortion system. The similarity to the Cartesian-loop linearisation scheme is obvious – the analogue differential amplifiers are replaced by the digital pre-distortion (DPD) block. Analogue-to-digital converters (ADCs) are needed for converting the demodulated output signal of the transmit path, containing the nonlinear power amplifier, to digital. The scheme presented in Fig. 1.3 is a signal pre-distortion scheme, where the transmit-signal, immediately before the conversion to analogue, is pre-distorted. The initially proposed data pre-distortion approach [3] pre-distorts the transmit data symbols. Signal pre-distortion requires even faster operation of the processing devices than in data pre-distortion systems, since the signal data-rate is in general higher than the symbol rate. Signal pre-distortion is further independent of the used modulation scheme, whereas in data pre-distortion the pre-distortion algorithm depends on the modulation format.

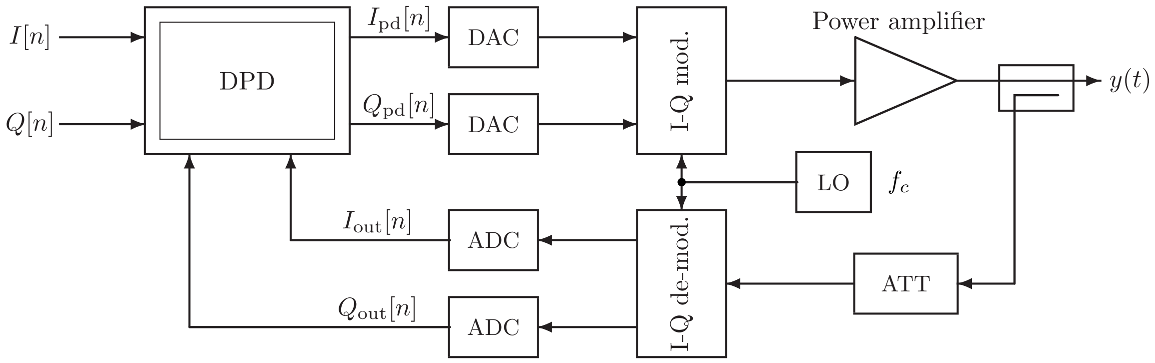


Figure 1.3: Digital pre-distortion linearisation scheme

Digital Pre-distortion: Building Blocks

Fig. 1.4 shows a detailed view of the building blocks of the digital pre-distortion system and their interaction. The components $I[n], Q[n]$ of the undistorted transmit-signal pass the pre-equalisation filter which distorts the signal in such a way that after passing the nonlinear analogue circuitry the signal appears linearly amplified. In order to achieve this linear amplification, the pre-equaliser has to be designed appropriately. The necessary knowledge comes from the modelling step which creates an accurate model of the nonlinear transmit circuitry. The analogue feedback-path must obey stringent linearity requirements, as in the case of the Cartesian-loop linearisation scheme. Thus, in a pre-distortion system the following tasks have to be performed:

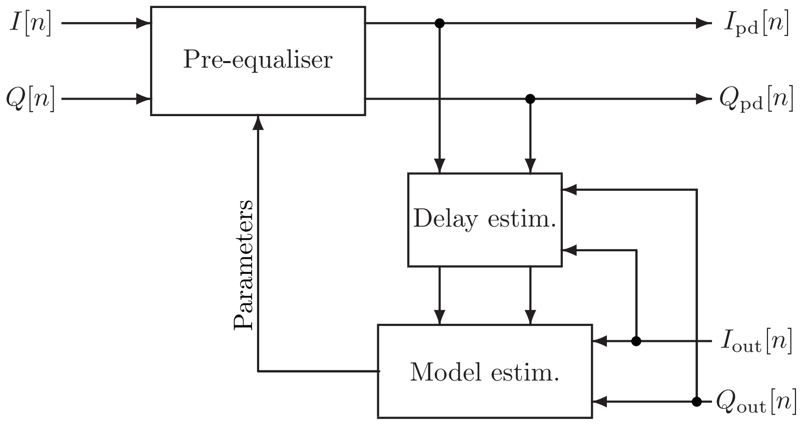


Figure 1.4: Digital pre-distortion linearisation scheme – building blocks

1. Modelling of the power amplifier (addressed in Chapter 2)

2. Adaptive identification of the power amplifier model-parameters (addressed in Chapter 3)
3. Design of the pre-distortion filter (addressed in Chapter 4)
4. Realisation of the pre-distortion unit (addressed in Chapter 5).

The first task is the modelling of the nonlinear transmit path. Nonlinear systems are basically all systems which behave nonlinear. Infinitely many possibilities for models exist, making this task difficult. A model has to be found that is capable of describing the behaviour of a composite of systems (mixed-signal devices such as DACs, analogue circuitry such as I-Q modulators, filters, pre-amplifiers, and power amplifiers) and that is at the same time reasonably low-complex.

Once a candidate model has been found the model-parameters have to be estimated. An adaptive identification algorithm, in its nature an on-line optimisation method, can perform this task.

The next task is then to design a pre-equaliser that compensates the nonlinear and dynamic effects (memory effects) of the transmit path. Analytic solutions for the pre-equaliser are in general not achievable. At least a method which provides an approximate solution for the pre-equaliser has to be found. It would be further desirable that this method provides solutions for the pre-equaliser for a variety of power amplifier models.

Finally, the system has to be realised and tested in an experimental setup. This is not only a proof-of-principle, but shows also the technical feasibility of the method under technological constraints.

Digital Pre-distortion – Brief Literature Review

Digital pre-distortion of microwave power amplifiers is a relatively young technique, initiated in the early 1980s with the paper of A. A. M. Saleh and J. Salz [3]. This and other early contributions consider data pre-distortion, i.e., the data symbols are distorted, not the transmit signal after transmit filtering. The pulse-shaping is thus performed after the pre-distortion stage. The spectral broadening due to the nonlinear amplifier cannot be avoided, but the nonlinear distortion of the data is compensated. These contributions consider nonlinear memoryless power amplifiers.

Data pre-distortion considering also memory effects appear in the late 1980s [32, 33, 34, 35], using Volterra filters as models for the nonlinear channel and for the pre-equaliser [36]. Nonlinear equalisation at the receiver, compensating for a nonlinear travelling wave tube amplifier and a linear channel, are considered in [37], whereas linearisation at the satellite transmitter is considered in [32], making the task easier since the amplifier can be considered as memoryless and the channel, which makes post-equalisation complicated, has no impact.

Pre-distortion of the transmit signal after the transmit filters appeared in the late 1980s [38, 39, 40], introducing a look-up table based pre-distortion concept, based on a memoryless nonlinear power amplifier model. Since then, a vast amount of literature has been published, based on memoryless as well as dynamic models for the power amplifier, see e.g., [41, 17, 42, 43], mostly based on computer simulations.

Chapter 2

Power Amplifier Modelling

The modelling task in the context of power amplifier pre-distortion is essential. Without knowledge of the system behaviour at a relatively high level of abstraction – in digital pre-distortion, parametric black-box models are used – equalisation will be an impossible challenge. The quality of the modelling has a great influence on the quality of the equalisation, since the pre-distortion unit is built based on the used power amplifier model, as observed in detail in Chapter 4. Therefore, modelling is a key element for pre-distortion techniques.

Models used commonly in power amplifier design and analysis range from detailed physical descriptions of the active device, the transistor, e.g., in power amplifier design, to relatively abstract models, incorporating equivalent circuits, e.g., for power amplifier characterisation. Here, the most abstract approach will be taken – the power amplifier is modelled as a parametric black box[1]. This brings several advantages, but has its drawbacks, too:

- Modelling at a high level of abstraction does not require specific and detailed knowledge of the functionality of the power amplifier. On the other hand, specific knowledge can improve the quality of the modelling.
- Gaining physical insight into system behaviour from abstract black box models is rarely possible. In the context of pre-distortion the aim is not to gain physical insight into the operation of power amplifiers.
- If performed in a smart way, black box models of complex dynamic systems can lead to compact descriptions. Few parameters can suffice to characterise a complex system with a large number of interconnected subsystems. This is especially important in system simulation – fast and accurate simulators can be realised.

The most often used approach to deal with a nonlinear system is its linearisation around a certain operating point; in other words, the nonlinear aspects of the problem are avoided. Approximating the power amplifier behaviour in some neighbourhood of an operating point using a linear model is not feasible in the pre-distortion context – the identification of the nonlinear aspects is essential for being able to compensate them.

In this chapter parametric nonlinear models for power amplifiers are presented. The Volterra series [45] is presented – it is the most widely investigated model for nonlinear dynamic systems. As it is capable of modelling a very large class of systems, it serves as a

---

[1] A model-set whose parameters are a vehicle for adjusting the fit to the data is called a black box [44]. The parameters do not reflect physical entities in the system. Accordingly, model-sets with adjustable parameters admitting physical interpretations are called grey boxes.

performance measure for the modelling capabilities of low complex models, such as Wiener- and Hammerstein models, which are presented thereafter. It is not the scope of this work to present a detailed analysis of the capabilities and difficulties associated with each presented model. The modelling is viewed from a practically relevant position, mostly with respect to performance and complexity.

The modelling capabilities of each model structure are tested against measured input/output data from power amplifiers, as well as against data obtained with a widely used simulation tool, ADS [2]. This simulation tool incorporates vendor-specific power amplifier models at circuit level. The advantage by using such an environment is that no equipment limitations, e.g., with respect to a maximum allowed signal bandwidth, have to be taken into consideration, as well as detrimental effects, such as measurement noise and accuracy limitations (e.g. limited resolution of analogue-to-digital converters), have no effect on the modelling. This environment is suited to test different model structures against each other and to gain insight in algorithmic limitations by excluding undesired effects from the measurement process. On the other hand, simulations can provide only a very special and restricted picture of reality, where detrimental effects and limitations have to be taken into account.

In order to prevent wrong conclusions on power amplifier models based only on simulated data, measurements on microwave power amplifiers have been performed. Therefore, the presented models in the following sections are compared with respect to their modelling capabilities, i.e., the capability to reproduce the measured output data given the input data.

2.1 The Volterra Series

The Volterra series as a means for describing nonlinear systems was first used by Norbert Wiener [47, 4]. It is a functional power series of the form

$$y(t) = \mathbb{H}\big(x(t)\big) = h_0 + \sum_{p=1}^{\infty} \int \cdots \int h_p(t, \tau_1, \ldots, \tau_p) x(\tau_1) \cdots x(\tau_p) d\tau_1 \cdots d\tau_p \ , \tag{2.1}$$

in which $\mathbb{H}\big(x(t)\big)$ is a nonlinear functional of the continuous function $x(t)$, h_0 is a constant, t is a parameter, and $h_p(\cdots)$ for $p \geq 1$ are continuous functions, called the Volterra kernels.

In the context of modelling of a dynamic nonlinear system such as a power amplifier in wireless communications, $x(t)$ is the input time-signal and t is (continuous) time. The basic questions are whether it is possible to approximate the behaviour of a physical nonlinear dynamic system with a series of the form in (2.1) and for which systems and which input signals does this series converge. These questions can be answered affirmatively for a large class of systems, i.e., any time-invariant continuous nonlinear system, and a wide class of input signals, i.e., signals extending over a finite time interval and belonging to a compact set[3] [5, pp. 34-37]. The basis for proving this is the Stone-Weierstrass theorem, see [49].

---

[2] ADS [46] stands for Advanced Design System, a simulation environment developed by Agilent Technologies. It is capable to simulate microwave designs, as well as complex communication systems.

[3] A set S in a normed space is compact, if for an arbitrary sequence $\{x_i\}$ in S there is a subsequence $\{x_{i_n}\}$ converging to an element $x \in S$. In finite dimensions, compactness is equivalent to being closed and bounded [48]. An example of a noncompact set is the unit ball in $L_2(0,T)$, the set of all $u(t)$ such that $\|u(t)\| \leq 1$, easily verified by an example, violating Weierstrass' theorem [48] which states that a continuous functional on a compact subset of a normed vector space achieves a maximum: The continuous functional $f(u) = \int_0^{1/2} u(t)dt - \int_{1/2}^{1} u(t)dt$ does not achieve the supremum of 1 ($|f(u)| \leq \|u(t)\| \leq 1$) for continuous functions $u(t)$, thus proves that the unit ball is not compact.

In [8] the range of the input signals for which the Volterra series approximation converges is enlarged to signals extending over the whole time-axis and belonging to non-compact sets, to which more practical signals belong to. This is achieved by limiting the class of systems to only those with fading memory. Roughly speaking, this means that the memory of the system "fades" or that the system is "forgetting". These strong theoretical results provide the background for the attempts in representing nonlinear dynamic systems with Volterra series.

The input/output relation of the Volterra model of the power amplifier used in this work is a truncated and stationary (time-invariant) Volterra series

$$\tilde{y}(t) = \sum_{p=1}^{2P+1} \int \cdots \int \tilde{h}_p(\tau_1, \ldots, \tau_p) \tilde{x}(t-\tau_1) \cdots \tilde{x}(t-\tau_p) d\tau_1 \cdots d\tau_p \tag{2.2}$$

$$= \sum_{p=1}^{2P+1} \int \tilde{h}_p(\boldsymbol{\tau}_p) \prod_{i=1}^{p} \tilde{x}(t-\tau_i) d\boldsymbol{\tau}_p \ , \tag{2.3}$$

where, for notational compactness, the vector $\boldsymbol{\tau}_p = [\tau_1, \ldots, \tau_p]^T$ for the set of time-arguments for the $p-$dimensional kernel $\tilde{h}_p(\tau_1, \ldots, \tau_p)$ and $d\boldsymbol{\tau}_p = d\tau_1 \cdots d\tau_p$ is used. The multiple integrals are compactly described. The constant term h_0 is assumed to be zero. The tilde marks the signals and the kernels as real-valued bandpass, see (2.4) further ahead. In the following, an equivalent discrete-time complex baseband input/output representation of (2.2) is deduced. This follows the derivation in [50], where the equivalent complex baseband Volterra kernels of a real-valued bandpass Wiener model are determined, see also Section 2.2 for more details.

2.1.1 Complex Baseband Volterra Series

During the following analysis it is assumed that the output of the power amplifier is filtered by a so called zonal filter $\tilde{g}(t)$ as depicted in Fig. 2.1. This filter selects the spectral zone of interest – here, the filter is tuned to the centre-frequency f_0 of the transmitter. The bandwidth is as large as necessary in order to cover nonlinear spectral broadening. In this analysis, the zonal filter is virtual and only necessary to compute the equivalent baseband Volterra-model – in a pre-distortion system, the filter exists and is incorporated in the mixer-stage or I/Q-demodulator, used for the conversion of the output signal of the power amplifier to baseband or to a lower intermediate frequency, see Fig. 1.3. It is assumed further, that the filter is perfectly flat in the selection-zone and with a perfect spectral suppression outside the selected zone. In a realised system the bandwidth is equivalent to the bandwidth of the mixer stage. Amplitude and phase distortions are present and limit the accuracy. Since this filter is in the feedback path of the pre-distortion system, deviations from the ideal assumptions have great influence on the overall behaviour.

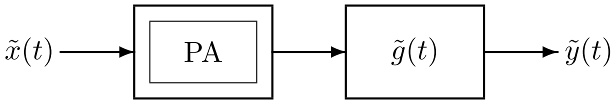


Figure 2.1: Power amplifier with zonal filter $\tilde{g}(t)$ at its output

The real-valued bandpass signal, centred at f_0 and input to the amplifier or Volterra model, is

$$\tilde{x}(t) = \Re\left\{x(t)e^{j2\pi f_0 t}\right\} \ . \tag{2.4}$$

The equivalent complex baseband signal is $x(t)$. Inserting the expression (2.4) into the first order term in the Volterra series (2.2) gives the output of the first order (linear) term

$$\tilde{y}_1(t) = \int \tilde{h}_1(\tau_1)\Re\left\{x(t-\tau_1)e^{j2\pi f_0(t-\tau_1)}\right\} d\tau_1 = \Re\left\{\int h_1(\tau_1)x(t-\tau_1)d\tau_1 e^{j2\pi f_0 t}\right\} \tag{2.5}$$

with the first order baseband kernel $h_1(t) = \tilde{h}_1(t)e^{-j2\pi f_0 t}$. The (ideal) zonal filter is centred at f_0, the signal passes unchanged. The equivalent complex baseband output signal is therefore simply obtained by the convolution of $h_1(t)$ and $x(t)$.

For the quadratic part insertion of (2.4) into

$$\tilde{y}_2(t) = \int \tilde{h}_2(\boldsymbol{\tau}_2)\prod_{i=1}^{2}\tilde{x}(t-\tau_i)d\boldsymbol{\tau}_2 \tag{2.6}$$

yields already a much more complex behaviour:

$$\begin{aligned}
\tilde{y}_2(t) &= \int \tilde{h}_2(\boldsymbol{\tau}_2)\prod_{i=1}^{2}\Re\left\{x(t-\tau_i)e^{j2\pi f_0(t-\tau_i)}\right\} d\boldsymbol{\tau}_2 \\
&= \frac{1}{4}\int \tilde{h}_2(\boldsymbol{\tau}_2)e^{-j2\pi f_0\tau_1}e^{-j2\pi f_0\tau_2}x(t-\tau_1)x(t-\tau_2)d\boldsymbol{\tau}_2 e^{j2\pi 2f_0 t} \\
&+ \frac{1}{4}\int \tilde{h}_2(\boldsymbol{\tau}_2)e^{j2\pi f_0\tau_1}e^{j2\pi f_0\tau_2}x^*(t-\tau_1)x^*(t-\tau_2)d\boldsymbol{\tau}_2 e^{-j2\pi 2f_0 t} \\
&+ \frac{1}{4}\int \tilde{h}_2(\boldsymbol{\tau}_2)e^{j2\pi f_0\tau_1}e^{-j2\pi f_0\tau_2}x^*(t-\tau_1)x(t-\tau_2)d\boldsymbol{\tau}_2 \\
&+ \frac{1}{4}\int \tilde{h}_2(\boldsymbol{\tau}_2)e^{-j2\pi f_0\tau_1}e^{j2\pi f_0\tau_2}x(t-\tau_1)x^*(t-\tau_2)d\boldsymbol{\tau}_2 \\
&= \frac{1}{2}\Re\left\{\int \tilde{h}_2(\boldsymbol{\tau}_2)e^{-j2\pi f_0(\tau_1+\tau_2)}\prod_{i=1}^{2}x(t-\tau_i)d\boldsymbol{\tau}_2 e^{j2\pi 2f_0 t}\right\} \\
&+ \frac{1}{2}\Re\left\{\int \tilde{h}_2(\boldsymbol{\tau}_2)e^{-j2\pi f_0(\tau_1-\tau_2)}x(t-\tau_1)x^*(t-\tau_2)d\boldsymbol{\tau}_2\right\} \ .
\end{aligned} \tag{2.7}$$

The second order part produces signals that are centred at $f = 2f_0$ and $f = 0$. The zonal filter, centred at f_0 with a bandwidth assumed to be small compared with f_0, suppresses the output of the second order term completely.

In a similar manner, the output signal of the third order part of (2.2) can be computed. The resulting expression is

$$\begin{aligned}
\tilde{y}_3(t) &= \frac{1}{4}\Re\left\{\int \tilde{h}_3(\boldsymbol{\tau}_3)e^{-j2\pi f_0(\tau_1+\tau_2+\tau_3)}\prod_{i=1}^{3}x(t-\tau_i)\,d\boldsymbol{\tau}_3\,e^{j2\pi 3f_0 t}\right\} \\
&+ \frac{1}{4}\Re\left\{\int \tilde{h}_3(\boldsymbol{\tau}_3)e^{-j2\pi f_0(\tau_1+\tau_2-\tau_3)}\ \ x(t-\tau_1)x(t-\tau_2)x^*(t-\tau_3)\,d\boldsymbol{\tau}_3\,e^{j2\pi f_0 t}\right\} \\
&+ \frac{1}{4}\Re\left\{\int \tilde{h}_3(\boldsymbol{\tau}_3)e^{-j2\pi f_0(\tau_1-\tau_2+\tau_3)}\ \ x(t-\tau_1)x^*(t-\tau_2)x(t-\tau_3)\,d\boldsymbol{\tau}_3\,e^{j2\pi f_0 t}\right\} \\
&+ \frac{1}{4}\Re\left\{\int \tilde{h}_3(\boldsymbol{\tau}_3)\,e^{-j2\pi f_0(-\tau_1+\tau_2+\tau_3)}x^*(t-\tau_1)x(t-\tau_2)x(t-\tau_3)\,d\boldsymbol{\tau}_3\,e^{j2\pi f_0 t}\right\} \ .
\end{aligned} \tag{2.8}$$

Only symmetric kernels $\tilde{h}_p(\mathbf{t}_p) = \tilde{h}_p(t_1, t_2 \ldots, t_p)$ are considered – the arguments $t_1, t_2, \ldots, t_p$ can therefore be permuted in every order without affecting the output signal. This is no loss of generality since every asymmetric Volterra kernel can easily be converted into a symmetric kernel, see [4, pp. 80-81]. The three last parts of (2.8) at f_0 yield therefore the same result. Since the zonal filter is centred at f_0, the first component at $3f_0$ is assumed to be suppressed perfectly. The only remaining term, written in the equivalent baseband form, is thus

$$y_3(t) = \int h_3(\boldsymbol{\tau}_3) x(t-\tau_1) x(t-\tau_2) x^*(t-\tau_3) d\boldsymbol{\tau}_3 \tag{2.9}$$

with the third order baseband kernel

$$h_3(\mathbf{t}_3) = \frac{3}{4} \tilde{h}_3(\mathbf{t}_3) e^{-j2\pi f_0 (t_1+t_2-t_3)} \ . \tag{2.10}$$

By induction it can be shown that the terms of even order $2p$ do not contribute to the output signal if the zonal filter is centred at f_0. Only the uneven terms in (2.2) produce components at f_0. The output for the $2p+1$-th homogeneous part is (in the equivalent baseband form)

$$y_{2p+1}(t) = \int h_{2p+1}(\boldsymbol{\tau}_{2p+1}) \prod_{i=1}^{p+1} x(t-\tau_i) \prod_{i=p+2}^{2p+1} x^*(t-\tau_i) d\boldsymbol{\tau}_{2p+1} \tag{2.11}$$

with the equivalent baseband kernel

$$h_{2p+1}(\mathbf{t}_{2p+1}) = \left(\frac{1}{2}\right)^{2p} \binom{2p+1}{p} \tilde{h}_{2p+1}\left(\mathbf{t}_{2p+1}\right) e^{-j2\pi f_0 \left(\sum_{i=1}^{p+1} t_i - \sum_{i=p+2}^{2p+1} t_i\right)} \ , \tag{2.12}$$

$\tilde{h}_{2p+1}\left(\mathbf{t}_{2p+1}\right)$ being the real-valued $2p+1$-dimensional bandpass kernel.

The equivalent complex baseband Volterra series of order $2P-1$ (with the zonal filter centred at the centre frequency f_0) is thus

$$y(t) = \sum_{p=0}^{P-1} \int h_{2p+1}(\boldsymbol{\tau}_{2p+1}) \prod_{i=1}^{p+1} x(t-\tau_i) \prod_{i=p+2}^{2p+1} x^*(t-\tau_i) d\boldsymbol{\tau}_{2p+1} \ . \tag{2.13}$$

It has to be noted that in (2.12) the equivalent baseband kernel $h_{2p+1}(\mathbf{t}_{2p+1})$ is invariant with respect to a permutation of the first $p+1$ arguments and with respect to a permutation of the second p arguments. A permutation of members between these two sets is allowed only if the corresponding conjugation of the input signal $x(t)$ in (2.13) is also considered.

2.1.2 Frequency Domain Representation of a Volterra Series

Further insight into the behaviour of a nonlinear system, represented by the Volterra series (2.13), can be gained if the output spectrum is computed. In the following, this calculation is performed exemplarily for the third order part (see [4, pp.104-108] for the real valued case). The extension to arbitrary orders is straightforward.

Consider (2.9), the third order homogeneous part. For the calculation of

$$y_{(3)}(t_1, t_2, t_3) = \int h_3(\boldsymbol{\tau}_3) x(t_1-\tau_1) x(t_2-\tau_2) x^*(t_3-\tau_3) d\boldsymbol{\tau}_3 \tag{2.14}$$

is introduced. Three dimensional Fourier transform yields

$$Y_{(3)}(f_1,f_2,f_3) = H_3(f_1,f_2,f_3)X(f_1)X(f_2)X^*(-f_3)\ . \tag{2.15}$$

Inverse Fourier transformation of (2.15) and setting of $t = t_1 = t_2 = t_3$ gives

$$y_3(t) = y_{(3)}(t,t,t) = \iiint H_3(f_1,f_2,f_3)X(f_1)X(f_2)X^*(-f_3)e^{j2\pi(f_1+f_2+f_3)t}df_1df_2df_3\ . \tag{2.16}$$

With substitution of $f_2 + f_3 = \nu_1$

$$y_3(t) = \iiint H_3(f_1,\nu_1-f_3,f_3)X(f_1)X(\nu_1-f_3)X^*(-f_3)e^{j2\pi(f_1+\nu_1)t}df_1d\nu_1df_3 \tag{2.17}$$

is obtained. Further substitution of $f = f_1 + \nu_1$ results in

$$y_3(t) = \iiint H_3(f-\nu_1,\nu_1-f_3,f_3)X(f-\nu_1)X(\nu_1-f_3)X^*(-f_3)e^{j2\pi ft}d\nu_1df_3df\ , \tag{2.18}$$

which provides the desired result for the spectrum $Y_3(f)$

$$Y_3(f) = \iint H_3(f-\nu_1,\nu_1-\nu_2,\nu_2)X(f-\nu_1)X(\nu_1-\nu_2)X^*(-\nu_2)d\nu_1d\nu_2\ . \tag{2.19}$$

For the general case of the $2p + 1$-th homogeneous part it can easily be shown that

$$Y_{2p+1}(f) = \int Y_{(2p+1)}(f-\nu_1,\nu_1-\nu_2,\nu_2-\nu_3,\ldots,\nu_{2p})d\nu_1\cdots d\nu_{2p}\ , \tag{2.20}$$

with

$$Y_{(2p+1)}(f_1,\ldots,f_{2p+1}) = H_{2p+1}(f_1,\ldots,f_{2p+1})\prod_{i=1}^{p+1}X(f_i)\prod_{i=p+2}^{2p+1}X^*(-f_i)\ . \tag{2.21}$$

The frequency domain representation of the Volterra series (2.13) is therefore

$$Y(f) = \sum_{p=0}^{P-1}Y_{2p+1}(f)\ . \tag{2.22}$$

Equation (2.20) is very similar to a multi-dimensional convolution integral. From this integral form the spectral broadening of a signal passing through a Volterra system can easily be understood. This becomes very clear in the case of a Wiener- or Hammerstein system, considered in Section 2.2 and Section 2.3 in more detail.

2.1.3 Discrete-time Volterra Series

A discrete-time representation of the Volterra series (2.13), used to model the nonlinear system, is essential for digital signal processing. In digital pre-distortion, the output of the amplifier (after attenuation, down-conversion and eventual demodulation) is sampled. This gives a discrete-time signal, together with the discrete-time input signal it is used to extract the parameters of the power amplifier model, here, the Volterra kernels. The selection of the sampling rate is essential – in order to reconstruct the analogue output signal, the sampling rate must be at least twice the signal bandwidth, see e.g., [51, 52], as requested by the sampling theorem. Since nonlinear systems spread the signal in bandwidth, see (2.20), selecting the sampling rate twice the output bandwidth is challenging in practice, since very fast analogue-to-digital converters with a high resolution (typically 12-14 bits supporting a dynamic range of up to 70 dB) have to be used. For system identification it can be shown that it is sufficient to sample the output with the same rate as used for the input signal [53, 54, 55].

The input signal $X(f)$ is assumed to be bandlimited[4] in $I = [-B, B]$. As can be seen from (2.21), the kernels $H_{2p+1}(f_1, \ldots, f_{2p+1}), p = 0, \ldots, P-1$, can be assumed to be bandlimited. The form of the kernels outside of the hypercube $C = I \times I \times \ldots \times I$ is of no importance, since in this region the kernels will not be excited by the input signal. The spectrum $Y_{(2p+1)}(f_1, \ldots, f_{2p+1})$ is zero outside the hypercube C and therefore strictly bandlimited[5].

It is assumed that the input signal is sampled at the Nyquist-rate $T = \frac{1}{2B}$ yielding $x[n] = x(nT) = x(\frac{n}{2B}), n \in \mathbb{Z}$. Discrete Fourier transform yields

$$X(f) = \sum_{n \in \mathbb{Z}} x[n] e^{-j\pi \frac{f}{B} n} \, . \tag{2.23}$$

Since the kernels can be assumed to be bandlimited, the spectrum of the kernel in the hypercube C can be determined by the multidimensional Fourier transform, e.g., for the third order kernel

$$H_3(f_1, f_2, f_3) = \sum_{n_1, n_2, n_3 \in \mathbb{Z}} h_3[n_1, n_2, n_3] e^{-j\frac{\pi}{B}(f_1 n_1 + f_2 n_2 + f_3 n_3)} \, , \tag{2.24}$$

the Volterra kernel being sampled at the Nyquist-rate of the input signal, $h_3[n_1, n_2, n_3] = h_3(\frac{n_1}{2B}, \frac{n_2}{2B}, \frac{n_3}{2B})$. If the output is sampled, again with the rate of the input sampling $y_3[n] = y(\frac{n}{2B})$, with (2.16), (2.23) and (2.24)

$$\begin{aligned}
y_3[n] &= \sum_{n_1, n_2, n_3 \in \mathbb{Z}} \sum_{r_1, r_2, r_3 \in \mathbb{Z}} h_3[n_1, n_2, n_3] x[r_1] x[r_2] x^*[r_3] \times \\
&\quad \frac{1}{2B} \int_{-B}^{B} e^{j\pi \frac{f_1}{B}(n - n_1 - r_1)} df_1 \frac{1}{2B} \int_{-B}^{B} e^{j\pi \frac{f_2}{B}(n - n_2 - r_2)} df_2 \frac{1}{2B} \int_{-B}^{B} e^{j\pi \frac{f_3}{B}(n - n_3 - r_3)} df_3 \\
&= \sum_{\mathbf{n}_3, \mathbf{r}_3} h_3[\mathbf{n}_3] x[r_1] x[r_2] x^*[r_3] \delta[n - n_1 - r_1] \delta[n - n_2 - r_2] \delta[n - n_3 - r_3] \\
&= \sum_{\mathbf{n}_3} h_3[\mathbf{n}_3] x[n - n_1] x[n - n_2] x^*[n - n_3]
\end{aligned} \tag{2.25}$$

is obtained, showing that the output of the Volterra system, sampled with the Nyquist-rate of the input-signal, can be obtained by a convolution of the sampled input signal and the sampled Volterra kernel. In order to shorten the notation the argument vectors $\mathbf{n}_3 = [n_1, n_2, n_3]^T$ and $\mathbf{r}_3 = [r_1, r_2, r_3]^T$ are introduced. A summation with the here three dimensional argument vectors as indices stands for a three-fold summation with the entries of the argument vectors used as individual indices. This notation is used wherever appropriate. Higher order kernels

---

[4] It has to be emphasised that in a pre-distortion system the input signal to the power amplifier has been nonlinearly distorted by the pre-distortion filter. The bandwidth is therefore larger than the bandwidth of the undistorted input signal, e.g., $P \times 5$ MHz for one UMTS carrier and for a pre-distortion filter of nonlinear order P.

[5] It has to be noted that $H_{2p+1}(f_1, \ldots, f_{2p+1})$ cannot be strictly bandlimited due to causality. Here, it is assumed that the spectral components of the input signal are sufficiently small outside the interval $I = [-B, B]$. Therefore, the signal $Y_{(2p+1)}(f_1, \ldots, f_{2p+1})$ is assumed to be bandlimited in C.

are treated equivalently. The discrete-time Volterra series of order $2P-1$ is

$$y[n] = \sum_{p=0}^{P-1} \sum_{\mathbf{n}_{2p+1} \in \mathbb{Z}} h_{2p+1}[\mathbf{n}_{2p+1}] \prod_{i=1}^{p+1} x[n-n_i] \prod_{i=p+2}^{2p+1} x^*[n-n_i] \; . \tag{2.26}$$

The continuous-time kernel $h_{2p+1}(\mathbf{t}_{2p+1})$ and the discrete-time kernel $h_{2p+1}[\mathbf{n}_{2p+1}]$ are equivalent, since the same output-signal is reproduced exactly at the sampling points. The continuous-time Volterra system $\mathbb{V}^c$, defined by the continuous-time kernels, and the discrete-time Volterra system $\mathbb{V}^d$, defined by the sampled continuous-time kernels, are therefore equivalent. Hence, for system identification, sampling with the Nyquist-rate of the input-signal is sufficient for estimating the Volterra-kernels. But it is not possible to reconstruct the continuous-time output signal $y(t)$ from the discrete-time signal $y[n]$, resulting from a sampling with the Nyquist-rate of the input signal. Aliasing would occur, since $Y(f)$ is not bandlimited in $I = [-B, B]$, see (2.20). If the discrete-time Volterra kernels are known, the continuous-time kernels can be reconstructed, thus, with the knowledge of the continuous-time input signal $x(t)$, the continuous-time output signal $y(t)$ can be produced. Hence, the knowledge of the discrete-time kernel and the input signal, either in continuous-time or discrete-time, is sufficient to reproduce all signals. The commutative diagram [53]

$$\begin{tikzcd} x(t) \arrow[r, "\mathbb{V}^c"] \arrow[d, leftrightarrow] & y(t) \arrow[d] \\ x[n] \arrow[r, "\mathbb{V}^d"] & y[n] \end{tikzcd} \tag{2.27}$$

visualises this situation: Taking the path $x(t) \xrightarrow{\mathbb{V}^c} y(t) \longrightarrow y[n]$ is equivalent to taking the path $x(t) \longrightarrow x[n] \xrightarrow{\mathbb{V}^d} y[n]$.

The consequence is that for modelling a nonlinear system which can be represented by a Volterra series, it is not necessary to sample the output signal at its Nyquist rate – sampling with the Nyquist-rate of the input signal is sufficient for estimating the Volterra kernels.

2.1.4 Series Representation of a Static Non-linearity

From the Volterra series (2.26) it is straightforward to specialise to a static non-linearity. The kernels are assumed to vanish, except if all indices are equal to zero, $h_{2p+1}[\mathbf{n}_{2p+1}] = 0$ if $\mathbf{n}_{2p+1} \neq \mathbf{0}, p = 0, \dots, P-1$. The equivalent discrete-time baseband representation of a static non-linearity, represented by a power series, is therefore

$$y[n] = \sum_{p=0}^{P-1} h_{2p+1}[\mathbf{0}] x[n] \big|x[n]\big|^{2p} = x[n] \sum_{p=0}^{P-1} \theta_{2p+1} \big|x[n]\big|^{2p} = x[n] g_\theta\Big(\big|x[n]\big|\Big) \; . \tag{2.28}$$

The $2p+1$-th order kernel reduces to a simple (complex) scalar θ_p. A signal-dependent gain $g_\theta\big(|x[n]|\big)$ distorts the input signal. Since this gain is complex, it introduces amplitude and phase distortions which depend on the amplitude of the input signal. The amplitude distortion (AM-AM conversion) is

$$\frac{\big|y[n]\big|}{\big|x[n]\big|} = \Big|g_\theta\Big(\big|x[n]\big|\Big)\Big| \; , \tag{2.29}$$

whereas the phase distortion (AM-PM conversion) is

$$\arg\bigl(y[n]\bigr) - \arg\bigl(x[n]\bigr) = \arg\left(g_\theta\Bigl(\bigl|x[n]\bigr|\Bigr)\right) \; . \tag{2.30}$$

If instead of a power series a set of orthogonal polynomials $\bigl\{\phi_p(\cdot)\bigr\}$ with the real argument $\bigl|x[n]\bigr|$ is used, only even order polynomials will contribute,

$$y[n] = x[n] \sum_{p=0}^{P-1} \theta'_{2p+1}\,\phi_{2p}\Bigl(\bigl|x[n]\bigr|\Bigr) \; . \tag{2.31}$$

E.g., for $\bigl\{\phi_p(\cdot)\bigr\}$ the even Hermite polynomials can be used. Another possibility is to use linear splines, see [21, 43].

2.1.5 Parameter Estimation for the Volterra Model

The Volterra series is linear-in-parameters – standard least-squares techniques can therefore be used to estimate the kernels. The composition of the regression matrix is, due to the special form of (2.26), different from the standard real-valued Volterra series: the kernels are symmetric with respect to the first $p+1$ arguments, e.g., $h_3[0,1,1] = h_3[1,0,1]$ since these kernels are associated with the signal products $x[n]x[n-1]x^*[n-1]$ and $x[n-1]x[n]x^*[n-1]$, which are symmetric with respect to the first two components. The kernels are symmetric also with respect to the second p arguments. These terms are associated with the conjugated signals. The difference to the real-valued standard formulation of the Volterra series is that permutations across these two sets, i.e., the kernel arguments corresponding to the product of the first $p+1$ not conjugated signals and the kernel arguments corresponding to the product of the second p conjugated signals, are not allowed.

The kernels can be arranged in vectors, e.g., the third order kernel with a one-tap memory comprises six elements, $\mathbf{h}_3 = \bigl[h[0,0,0], h[0,0,1], h[0,1,0], h[0,1,1], h[1,1,0], h[1,1,1]\bigr]^T$, symmetries already taken into account. The parameter vector of a Volterra series of order $2P-1$, containing only uneven components, can therefore be written as

$$\boldsymbol{\theta} = [\mathbf{h}_1^T, \mathbf{h}_3^T, \ldots, \mathbf{h}_{2P-1}^T]^T \; . \tag{2.32}$$

A specific signal matrix $\mathbf{H}$, associated with this parameter vector, is composed of sub-matrices $\mathbf{X}_p$, which are associated with the kernels $\mathbf{h}_p$,

$$\mathbf{H} = [\mathbf{X}_1, \mathbf{X}_3, \ldots, \mathbf{X}_{2P-1}] \; . \tag{2.33}$$

Here, e.g., the sub-matrix $\mathbf{X}_3$, considering a time window of M samples, is

$$\mathbf{X}_3 = [\mathbf{x}_{3,n}, \mathbf{x}_{3,n-1}, \ldots, \mathbf{x}_{3,n-M}]^T \; , \tag{2.34}$$

whereby the vectors

$$\mathbf{x}_{3,n} = \Bigl[x[n]\bigl|x[n]\bigr|^2, x^2[n]x^*[n-1], \ldots, x[n-N_3+1]\bigl|x[n-N_3+1]\bigr|^2\Bigr]^T \; , \tag{2.35}$$

N_3 being the memory-length of the kernel, are used.

The output signal of a Volterra system of order $2P-1$, see (2.26), over the finite time-horizon $n, n-1, \ldots, n-M$ is thus

$$\mathbf{y}_n = \mathbf{H}\boldsymbol{\theta} \tag{2.36}$$

with $\mathbf{y}_n = \big[y[n], \ldots, y[n-M]\big]^T$.

Having available a set of measured output data $\mathbf{d}_n = \big[d[n], d[n-1], \ldots, d[n-M]\big]^T$, the parameters (kernels) of a specific Volterra model can be estimated with the least-squares method:

$$\hat{\boldsymbol{\theta}} = (\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H\mathbf{d}_n \; . \tag{2.37}$$

2.2 The Wiener Model

The Wiener model used here is a series connection of a linear finite-impulse response (FIR) filter and a static (memoryless) non-linearity $f(\cdot)$, see Fig. 2.2. It is not the general Wiener model, resulting from a construction of orthogonal functionals (Wieners G-functionals) from the Volterra functionals via a Gram-Schmidt procedure [4, pp. 199ff.]. The general Wiener model has the advantage to incorporate a larger class of nonlinear systems than a Volterra series, but is not less complex (in terms of numbers of parameters) as the Volterra series. The simple Wiener model [44] used here requires substantially fewer parameters compared to a Volterra model and achieves good modelling results, see Section 2.6.

As in the case of the Volterra model, at the output of the power amplifier a zonal filter selects the spectral zone around the carrier frequency. The linear filter in the Wiener model is assumed to be an FIR filter $\mathbb{G}(q^{-1}) = \sum_{i=0}^{N-1} g_i q^{-i}$, the nonlinear function is represented via a series as in (2.31). Hence, the discrete-time input/output representation of the Wiener model is

$$y[n] = \mathbb{G}\big(x[n]\big) \sum_{p=0}^{P-1} \theta_{2p+1}\, \phi_{2p}\left(\Big|\mathbb{G}\big(x[n]\big)\Big|\right) \; . \tag{2.38}$$

In contrast to the Volterra series (and the static nonlinear function represented by a series) the Wiener model is not linear with respect to the parameters $\{g_i\}_{i=0}^{N-1}$ of the linear filter, it is only linear with respect to the parameters of the static non-linearity $\{\theta_p\}_{p=0}^{P-1}$.

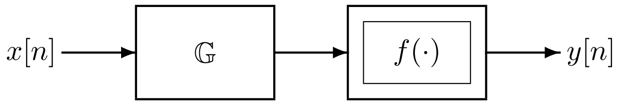


Figure 2.2: The Wiener model.

Further insight can be gained in the frequency domain. For this, the static non-linearity is represented using a power series. In continuous time, the input/output relation reads

$$y(t) = \sum_{p=0}^{P-1} \theta_{2p+1}\mathbb{G}\big(x(t)\big)\Big|\mathbb{G}\big(x(t)\big)\Big|^{2p} \; . \tag{2.39}$$

Exemplarily, the third order part is considered,

$$y_{(3)}(t_1, t_2, t_3) = \theta_3\mathbb{G}\big(x(t_1)\big)\mathbb{G}\big(x(t_2)\big)\mathbb{G}^*\big(x(t_3)\big) \; . \tag{2.40}$$

Fourier transform yields

$$\begin{aligned} Y_{(3)}(f_1, f_2, f_3) &= \theta_3 G(f_1)X(f_1)G(f_2)X(f_2)G^*(-f_3)X^*(-f_3) \\ &= \theta_3 H(f_1, f_2, f_3)X(f_1)X(f_2)X^*(-f_3) \\ &= \theta_3 Z(f_1)Z(f_2)Z^*(-f_3)\ , \end{aligned} \tag{2.41}$$

with $Z(f) = G(f)X(f)$. The Volterra kernel in the frequency domain is $H(f_1, f_2, f_3) = \theta_3 G(f_1)G(f_2)G^*(-f_3)$. Again, $X(f)$ is considered exactly bandlimited in $I = [-B, B]$. Therefore, $Y(f_1, f_2, f_3)$ is exactly bandlimited in the 3-dimensional cube $C = I \times I \times I$. Inverse Fourier transform yields

$$y(t) = y_{(3)}(t, t, t) = \theta_3 \iiint Z(f - \nu_1)Z(\nu_1 - \nu_2)Z^*(-\nu_2)d\nu_1 d\nu_2 e^{2\pi f t} df\ , \tag{2.42}$$

where $Y(f)$ can be recognised to be

$$Y(f) = \theta_3 \iint Z(f - \nu_1)Z(\nu_1 - \nu_2)Z^*(-\nu_2)d\nu_1 d\nu_2 = \theta_3 Z(f) * Z(f) * Z^*(-f)\ , \tag{2.43}$$

a twofold convolution of $Z(f) = H(f)X(f)$ with itself, producing the spectral broadening, $Y(f) \in [-3B, 3B]$. Via the convolution, the spectrum $Y_{(3)}(f_1, f_2, f_3) \in C$ is mapped into $Y(f) \in [-3B, 3B]$.

Again, as in the case of a Volterra system, for system identification input rate sampling at the output of the Wiener system suffices, i.e., even if the signal is spread in the frequency domain during the passage through the nonlinear system, no larger sampling rate at the output is required.

2.2.1 Parameter Estimation for the Wiener Model

The Wiener model is partially linear-in-parameters – the parameters of the static non-linearity $\{\theta_p\}$ – and partially nonlinear-in-parameters – the parameters of the linear filter $\mathbb{G}$. Parameter estimation using the conventional least-squares approach is therefore not applicable in a straightforward manner. For the parameter estimation with the measured and simulated data the following simple heuristic method is proposed:

1. First, the measured or simulated input/output data is fitted to a purely linear model using the least squares method:

   $$\hat{\mathbf{g}} = \left(\mathbf{H}_g^H \mathbf{H}_g\right)^{-1} \mathbf{H}_g^H \mathbf{d}_n\ , \tag{2.44}$$

   with the estimated parameter vector $\hat{\mathbf{g}} = [\hat{g}_0, \hat{g}_1, \ldots, \hat{g}_{N-1}]^T$, the input-signal matrix

   $$\mathbf{H}_g = \begin{pmatrix} x[n] & x[n-1] & \cdots & x[n-N+1] \\ x[n-1] & x[n-2] & \ldots & x[n-N] \\ \vdots & \vdots & \ddots & \vdots \\ x[n-M] & x[n-M-1] & \cdots & x[n-M-N+1] \end{pmatrix}, \tag{2.45}$$

   and $\mathbf{d}_n = \left[d[n], d[n-1], \ldots, d[n-M]\right]^T$, which is the measured output signal. The nonlinear distortions act here as an additional disturbance.

2. After the estimation of the parameters of the linear filter, the input signal is passed through this estimated linear filter, producing the input signal for the estimation of the static nonlinear filter
$$\hat{z}[n] = \hat{\mathbb{G}}\big(x[n]\big) \ . \tag{2.46}$$

3. Using this signal, the parameters of the nonlinear filter are estimated with the least squares technique
$$\hat{\boldsymbol{\theta}} = \big(\mathbf{H}_\theta^H \mathbf{H}_\theta\big)^{-1} \mathbf{H}_\theta^H \mathbf{d}_n \ , \tag{2.47}$$
with the estimated parameter vector $\hat{\boldsymbol{\theta}} = \big[\hat{\theta}_1, \hat{\theta}_3, \ldots, \hat{\theta}_{2P-1}\big]^T$, the signal matrix
$$\mathbf{H}_\theta = \begin{pmatrix} \hat{z}[n]\phi_0\big(|\hat{z}[n]|\big) & \cdots & \hat{z}[n]\phi_{2(P-1)}\big(|\hat{z}[n]|\big) \\ \hat{z}[n-1]\phi_0\big(|\hat{z}[n-1]|\big) & \ldots & \hat{z}[n-1]\phi_{2(P-1)}\big(|\hat{z}[n-1]|\big) \\ \vdots & \ddots & \vdots \\ \hat{z}[n-M]\phi_0\big(|\hat{z}[n-M]|\big) & \cdots & \hat{z}[n-M]\phi_{2(P-1)}\big(|\hat{z}[n-M]|\big) \end{pmatrix} , \tag{2.48}$$
and $\mathbf{d}_n = \big[d[n], d[n-1], \ldots, d[n-M]\big]^T$, which is again the measured output signal. In this last step a new data set is used, meaning that the input/output signals are not the same as those used in the estimation of the linear filter parameters.

If, e.g., $\phi_{2p}(x) = x^{2p}$,
$$y[n] = \theta_1 \mathbb{G}\big(x[n]\big) + \underbrace{\mathbb{G}\big(x[n]\big) \sum_{p=1}^{P-1} \theta_{2p+1} \Big|\mathbb{G}\big(x[n]\big)\Big|^{2p}}_{\text{nonlinear distortion}} \tag{2.49}$$
the nonlinear parts can be seen as a disturbance term for the estimation of the parameters of the linear filter. If the nonlinear disturbances are not too strong, which is the case for weakly nonlinear systems, the estimation of the parameters of the linear filter is accurate.

2.3 The Hammerstein Model

The Hammerstein model [44] investigated here is a static non-linearity $f(\cdot)$, followed by a linear FIR filter, see. Fig. 2.3. Also here, a zonal filter selects the spectral zone around the carrier frequency at the output of the power amplifier.

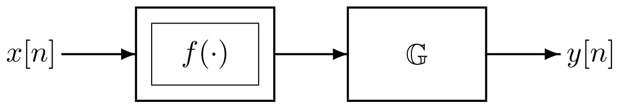


Figure 2.3: The Hammerstein model.

If the nonlinear function $f(\cdot)$ is represented by a series, the input/output representation reads
$$y[n] = \mathbb{G}\left(x[n] \sum_{p=0}^{P-1} \theta_{2p+1}\, \phi_{2p}\Big(\big|x[n]\big|\Big) \right) \ , \tag{2.50}$$

with the linear filter $\mathbb{G}(q^{-1}) = \sum_{i=0}^{N-1} g_i q^{-i}$. Similar as in the case of the Wiener model, the description is linear in a part of the parameters (the parameters of the linear filter), but nonlinear in the parameters of the static nonlinear function.

As in the case of the Wiener model, the continuous-time representation

$$y(t) = \mathbb{G}\left(\sum_{p=0}^{P-1} \theta_{2p+1} x(t) \big|x(t)\big|^{2p}\right) = \mathbb{G}\big(z(t)\big) \tag{2.51}$$

is transformed into frequency domain

$$Y(f) = G(f) Z(f) , \tag{2.52}$$

where, again only for the third-order term,

$$Z(f) = \theta_3 X(f) * X(f) * X^*(-f) . \tag{2.53}$$

Here, $Z(f) \in [-3B, 3B]$, $G(f)$ extends in general also over $[-3B, 3B]$, in contrast to the Wiener model, where the linear filter $G(f)$ can be considered bandlimited in $[-B, B]$.

2.3.1 Parameter Estimation for the Hammerstein Model

Similar as in the case of the Wiener model also the Hammerstein model is linear in part of the parameters – the parameters of the linear filter – and nonlinear in the parameters of the static nonlinear filter. A similar two-step estimation procedure is adopted also here:

1. The parameters of the static nonlinear filter are estimated first using the least squares method,
$$\hat{\boldsymbol{\theta}} = \left(\mathbf{H}_\theta^H \mathbf{H}_\theta\right)^{-1} \mathbf{H}_\theta^H \mathbf{d}_n , \tag{2.54}$$
whereby
$$\mathbf{H}_\theta = \begin{pmatrix} \phi_0\big(|x[n]|\big) & \cdots & \phi_{2(P-1)}\big(|x[n]|\big) \\ \phi_0\big(|x[n-1]|\big) & \ldots & \phi_{2(P-1)}\big(|x[n-1]|\big) \\ \vdots & \ddots & \vdots \\ \phi_0\big(|x[n-M]|\big) & \cdots & \phi_{2(P-1)}\big(|x[n-M]|\big) \end{pmatrix} , \tag{2.55}$$
$\hat{\boldsymbol{\theta}} = [\hat{\theta}_1, \ldots, \hat{\theta}_{2P-1}]^T$ is the estimated parameter vector and $\mathbf{d}_n = \big[d[n], \ldots, d[n-M]\big]^T$ is the measured output signal.

2. The input signal is then passed through the estimated nonlinear filter producing the new signal
$$\hat{z}[n] = f\big(\hat{\boldsymbol{\theta}}, x[n]\big) . \tag{2.56}$$

3. Using this signal as the input signal for the linear filter, the parameters of the linear filter are estimated, again using the least squares technique
$$\hat{\mathbf{g}} = \left(\mathbf{H}_g^H \mathbf{H}_g\right)^{-1} \mathbf{H}_g^H \mathbf{d}_n , \tag{2.57}$$
with the estimated parameter vector $\hat{\mathbf{g}} = [\hat{g}_0, \hat{g}_1, \ldots, \hat{g}_{N-1}]^T$, the input-signal matrix
$$\mathbf{H}_g = \begin{pmatrix} \hat{z}[n] & \hat{z}[n-1] & \cdots & \hat{z}[n-N+1] \\ \hat{z}[n-1] & \hat{z}[n-2] & \ldots & \hat{z}[n-N] \\ \vdots & \vdots & \ddots & \vdots \\ \hat{z}[n-M] & \hat{z}[n-M-1] & \cdots & \hat{z}[n-M-N+1] \end{pmatrix} , \tag{2.58}$$

and $\mathbf{d}_n = \big[d[n], d[n-1], \ldots, d[n-M]\big]^T$, which is the measured output signal. Also here, a new set the input and measured output data is used for the second step of the parameter estimation.

Equivalently as in the case of the Wiener model, the estimation of the first set of parameters (the parameters of the non-linearity) is not accurate due to the linear filter after that block. From

$$y[n] = \sum_{p=0}^{P-1} \theta_{2p+1} \mathbb{G}\Big(x[n]\big|x[n]\big|^{2p}\Big) = \tag{2.59}$$

$$g_0 \sum_{p=0}^{P-1} \theta_{2p+1} x[n]\big|x[n]\big|^{2p} + \underbrace{g_1 \sum_{p=0}^{P-1} \theta_{2p+1} x[n-1]\big|x[n-1]\big|^{2p} + \ldots}_{\text{memory effects}} \tag{2.60}$$

can be seen that the estimation of the parameters $\{\theta_p\}$ would be accurate if no dynamic part would exist, $g_1 = g_2 = \ldots = g_M = 0$. Since the memory lengths of the power amplifiers simulated and measured are not very long ($|g_0|$ is large compared to $|g_1|, |g_2|, \ldots$, see Section 2.6), the estimation of the parameters $\{\theta_p\}$ is not disturbed exceedingly.

2.4 The Saleh Model

A simple static power amplifier model requiring only four parameters is the Saleh model [9]. Originally, it is used to model traveling-wave tube amplifiers but it is often used to model solid-state power amplifiers, too. Both, amplitude distortions and phase distortions are modelled with simple two-parameter formulas.

If the (real-valued bandpass) input signal of the power amplifier is

$$\tilde{x}(t) = \Re\big\{x(t)e^{j2\pi f_0 t}\big\} = r(t)\cos\big(2\pi f_0 t + \psi(t)\big) \tag{2.61}$$

with $x(t) = r(t)e^{j\psi(t)}$ describing the equivalent low-pass signal, $r(t)$ being the amplitude and $\psi(t)$ being the phase of the this signal, the output signal of the power amplifier is modelled as

$$\tilde{y}(t) = A\big[r(t)\big]\cos\Big(2\pi f_0 t + \psi(t) + \Phi\big[r(t)\big]\Big) \tag{2.62}$$

with the nonlinear functions

$$A(r) = \frac{\alpha_a r}{1 + \beta_a r^2} \quad \text{and} \tag{2.63}$$

$$\Phi(r) = \frac{\alpha_\phi r^2}{1 + \beta_\phi r^2}\,. \tag{2.64}$$

If r is very large, $A(r)$ is proportional to $1/r$ and $\Phi(r)$ approaches a constant. The function $A(r)$ describes the conversion of the input amplitude to the output amplitude (AM-AM conversion), whereas the function $\Phi(r)$ describes the influence of the input amplitude on the output phase (AM-PM conversion).

2.4.1 Parameter Estimation for the Saleh Model

The equations (2.63) and (2.64) are reorganised for the estimation of the parameters $\alpha_a, \beta_a, \alpha_\phi$ and β_ϕ:

$$\frac{r[n]}{A[n]} = \frac{1}{\alpha_a} + \frac{\beta_a}{\alpha_a} r^2[n] \tag{2.65}$$

$$\frac{r^2[n]}{\Phi[n]} = \frac{1}{\alpha_\phi} + \frac{\beta_\phi}{\alpha_\phi} r^2[n] \; . \tag{2.66}$$

Here, $A[n] = A(r[n])$ and $\Phi[n] = \Phi(r[n])$ are the measured output amplitude and phase-difference, $r[n]$ is the amplitude of the input signal. The so transformed model allows to apply the standard least-squares technique for estimating the parameters, summarised in the parameter-vectors $\boldsymbol{\theta}_a = \left[\frac{1}{\alpha_a}, \frac{\beta_a}{\alpha_a}\right]^T$ and $\boldsymbol{\theta}_\phi = \left[\frac{1}{\alpha_\phi}, \frac{\alpha_\phi}{\beta_\phi}\right]^T$:

$$\hat{\boldsymbol{\theta}}_a = \left(\mathbf{H}_a^H \mathbf{H}_a\right)^{-1} \mathbf{H}_a^H \mathbf{w}_a \; , \tag{2.67}$$

$$\hat{\boldsymbol{\theta}}_\phi = \left(\mathbf{H}_\phi^H \mathbf{H}_\phi\right)^{-1} \mathbf{H}_\phi^H \mathbf{w}_\phi \; , \tag{2.68}$$

where

$$\mathbf{H}_a = \mathbf{H}_\phi = \begin{pmatrix} 1 & r^2[n] \\ 1 & r^2[n-1] \\ \vdots & \vdots \\ 1 & r^2[n-N+1] \end{pmatrix} \tag{2.69}$$

and

$$\mathbf{w}_a = \left[\frac{r[n]}{A[n]}, \frac{r[n-1]}{A[n-1]}, \ldots, \frac{r[n-N+1]}{A[n-N+1]}\right]^T \tag{2.70}$$

$$\mathbf{w}_\phi = \left[\frac{r^2[n]}{\Phi[n]}, \frac{r^2[n-1]}{\Phi[n-1]}, \ldots, \frac{r^2[n-N+1]}{\Phi[n-N+1]}\right]^T \; . \tag{2.71}$$

Once $\hat{\boldsymbol{\theta}}_a$ and $\hat{\boldsymbol{\theta}}_\phi$ are computed, $\hat{\alpha}_a = \frac{1}{\hat{\boldsymbol{\theta}}_a(1)}, \hat{\beta}_a = \frac{\hat{\boldsymbol{\theta}}_a(2)}{\hat{\boldsymbol{\theta}}_a(1)}, \hat{\alpha}_\phi = \frac{1}{\hat{\boldsymbol{\theta}}_\phi(1)}$, and $\hat{\beta}_\phi = \frac{\hat{\boldsymbol{\theta}}_\phi(1)}{\hat{\boldsymbol{\theta}}_\phi(2)}$ can easily be computed.

2.5 Model-Structure Selection and Model Validation

So far, different kinds of nonlinear models with various complexity have been introduced. The Volterra model is the most powerful model and comprises a large class of nonlinear dynamic systems. Wiener-and Hammerstein models are special cases of the Volterra series, as well as the static nonlinearities – the power series and Saleh's model. The models can be related to each other in terms of the size of the model class $\mathscr{M}$: The model class contains all models of a particular class, e.g., the class (set) $\mathscr{M}_{\text{Volterra}}$ embraces all models which can be expressed as a Volterra series, a particular model out of that class is determined by the parameter-vector $\boldsymbol{\theta}$ of a certain dimension $d_{\mathscr{M}(\boldsymbol{\theta})} = \dim \boldsymbol{\theta}$, which is a measure of the model-complexity. The described models can be ordered according to

$$\mathscr{M}_{\text{Saleh}} \subset \mathscr{M}_{\text{Power series}} \subset \mathscr{M}_{\text{Wiener}}, \mathscr{M}_{\text{Hammerstein}} \subset \mathscr{M}_{\text{Volterra}} \; . \tag{2.72}$$

The aim is to describe the input/output behaviour of the power amplifier with a sufficiently powerful model with the smallest dimension ($d_{\mathscr{M}(\boldsymbol{\theta})}$) possible. The evaluation criterion of choice for a particular model $m = \mathscr{M}(\boldsymbol{\theta})$ is the mean squared-error between system output $d[n]$ and model output $y_m[n]$

$$J(m) = \frac{1}{N}\sum_{n=1}^{N}\left|d[n] - y_m[n]\right|^2 = \frac{1}{N}\sum_{n=1}^{N}\left|e_m[n]\right|^2 , \tag{2.73}$$

where an observation interval of length N samples is chosen and system and model are excited with the same signal. The least-squares parameter estimation minimises this measure – but care has to be taken with respect to the number of parameters: using more parameters, the error will decrease monotonically. Unnecessary parameters will adjust themselves to the particular realisation of the noise (e.g., measurement noise). This is known as overfit – parameters which are adjusted to the specific noise realisation do not reflect features of the system and are thus useless.

2.5.1 Model-Structure Selection

Two standard criteria (tailored for least-squares parameter estimation and modifying the squared-error $J(m)$) for determining the model structure, i.e., the dimension of the model-parameter vector $\boldsymbol{\theta}$ of a model in a particular model class $\mathscr{M}$, are of the form [44]

$$W(m) = J(m)(1 + U(m)) , \tag{2.74}$$

where $W(m)$ is a modified least-squares error. The function $U(m)$ measures the dimension of the model structure and can be interpreted as a penalty-term. In Akaike's "Final Prediction-Error Criteria (FPE)" [56], the complexity-penalty is defined as

$$U_{\mathrm{FPE}}(m) = \frac{2}{N}\dim\boldsymbol{\theta} , \tag{2.75}$$

whereas in Rissanen's "Minimum Description Length Criteria (MDL)" [57], the data length is also taken into account

$$U_{\mathrm{MDL}}(m) = \frac{\ln(N)}{N}\dim\boldsymbol{\theta} . \tag{2.76}$$

This form of the penalty-term is obtained by requiring to achieve the shortest possible description of the observed data.

2.5.2 Model Validation

It is obvious that the derived model is capable of reproducing the data which has been used for the estimation. In order to test the model and being able to compare with other models, the models must be tested with different data sets – the validation data [44]. Comparing two different models m_1 and m_2 consists thus of computing $J(m_1)$ and $J(m_2)$, cf. (2.73), with a data set which is different from the data set used for the parameter estimation. The model which yields a better performance (smaller mean-squared error) and which complexity is reasonable for implementation, is favoured. The models are thus cross-validated against each other in terms of modelling capability and complexity. The comparison does not use any assumptions about the true system or other probabilistic arguments. The disadvantage of this method is the extra complexity of collecting and storing new data from the system, as well as evaluating the performance measure.

This approach is adopted in the following Section 2.6, where the validation data is different from the estimation data.

2.6 Modelling Measured and Simulated Power Amplifiers

In this section power amplifiers are modelled with the presented black-box models. The modelling is based on measurements from which input and output data is gathered. One set of data is generated using a commercially and widely used simulation tool – Advanced Design System (ADS) [46] – which generates the data from a circuit-based analogue power amplifier model, supplied by the manufacturer of the specific power amplifier. This is mainly included for comparison, a realistic measurement setup, although limited in the accuracy and the performance by the measurement equipment, is mandatory for obtaining the correct picture of reality.

For a first (qualitative) insight Fig. 2.4 shows the measured spectra of the input and output signal of a power amplifier (Mini-Circuits, ZVE-8G [58]). The power amplifier is excited with a multi-tone signal with a large number of tones. The crest-factor of the discrete-time multi-tone input signal $x[n]$ before digital-to-analogue conversion, defined as

$$\mathrm{CF}(x)\,[dB] = 20 \log \left(\frac{\left\| x[n] \right\|_\infty}{\left\| x[n] \right\|_2} \right) , \tag{2.77}$$

is approx. 5,4 dB. Here, the norms for a (discrete-time) periodic signal with period N are defined [59] as

$$\left\| x[n] \right\|_\infty \triangleq \sup_n \left| x[n] \right| \tag{2.78}$$

$$\left\| x[n] \right\|_2 \triangleq \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} \left| x[n] \right|^2} . \tag{2.79}$$

The bandwidth of the input signal is 5 MHz.

Spectral regrowth due to the non-linearity of the amplifier can be observed. The distance between the in-band spectral density and the out-of-band spectral density is in this case only approx. 20 dB. This cannot be tolerated in a technical application since neighbouring frequency bands are disturbed excessively. On the other hand, reducing the input power (often called input back-off – IBO), which gives rise to a more linear behaviour of the power amplifier, reduces the amplifier efficiency and is thus not a very economic solution.

Fig. 2.5(a) shows the magnitude of the measured output signal vs. the magnitude of the input signal, here referred to as the AM-AM conversion. The saturation behaviour can easily be recognised. Dispersive effects, which produce the broadening of the curve, are noticed. At this early point it has to be emphasised that in all cases where measurements on power amplifiers have been carried out, dispersive effects have very short time constants – two to three taps (e.g., in the linear filter of the Wiener model) or Volterra kernels of length two to three are in all cases adequate to model this effect. With a sampling rate of 100 MHz, two taps correspond to a memory of only 20 ns – this has to be compared with a chip-duration in WCDMA of approx. 260 ns, which yields a memory of only 7% of a chip. If approximated

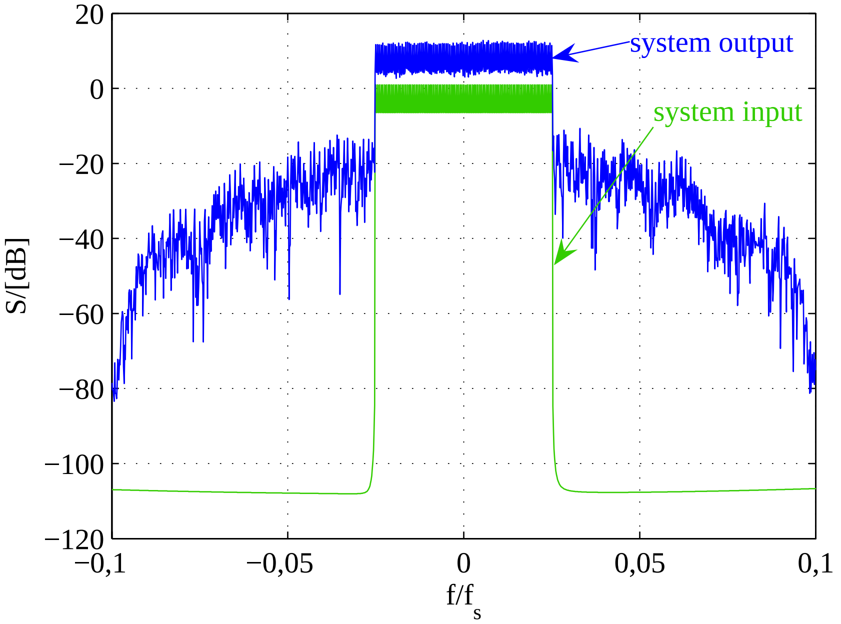


Figure 2.4: Measured spectra of a commercial power amplifier (Mini-Circuits ZVE-8G) – $f_s = 100\,\mathrm{MHz}$

by a linear system, the investigated amplifiers can be seen as spectrally flat (near constant amplitude response with a linear phase response), producing not significant dispersive effects. Other settings, e.g., by increasing the signal bandwidth significantly, are expected to enforce the usage of more taps in a Wiener model or longer kernels in a Volterra model.

Fig. 2.5(b) shows the effect of the amplitude of the input signal on the phase of the output signal, here referred to as the AM-PM conversion. The ordinate shows the difference of the phase of the output signal $\phi_d[n] = \arg\big(d[n]\big)$ and the phase of the input signal $\phi_x[n] = \arg\big(x[n]\big)$, $\Delta\phi[n] = \phi_d[n] - \phi_x[n]$, as a function of the input amplitude. Besides the memory effects, (which can be modelled with very short kernels) no significant dependence of the phase from the magnitude of the input signal can be observed.

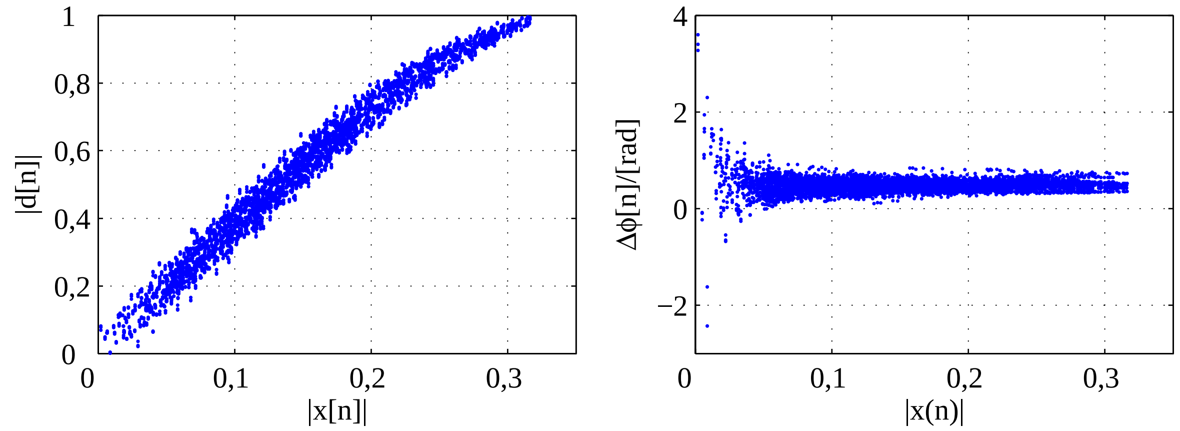
2.6.1 Black-Box Modelling of Three Microwave Power Amplifiers

Three microwave power amplifiers are modelled using the presented models. A single stage medium-power amplifier and a high-power amplifier with three stages are modelled based on measured input/output data. The high-power amplifier is an LDMOS[6] EDGE[7] amplifier. The maximum output power of the high-power amplifier is approx. 80 W in continuous wave operation (single carrier). The modelled single-stage amplifier is a medium-power amplifier (Mini-Circuits ZVE-8G, see [58]) with a maximum output power of only 1 W.

---

[6] LDMOS stands for Laterally Diffused Metal on Silicon (LDMOS) Field Effect Transistors (FET), a technology that is used for high-gain and high-linearity power amplifiers.

[7] EDGE (or Enhanced Data Rates for Global Evolution) is a 3G technology that delivers broadband-like data speeds to mobile devices. On the physical layer, EDGE only introduces a new modulation technique (8PSK-GMSK) and new channel coding that can be used to transmit both packet-switched and circuit-switched voice and data services.



(a) AM-AM conversion

(b) AM-PM conversion

Figure 2.5: Measured AM-AM and AM-PM characteristic of a commercial power amplifier (Mini-Circuits ZVE-8G)

Further, an LDMOS high power amplifier for application in WCDMA is modelled based on input/output data generated by a simulation environment which uses a circuit-based model of the amplifier.

The performance measure of the modelling is the mean-squared error (2.73), now normalised with the mean power of the output data

$$J_r(m)[dB] = 10\log\left(\frac{J(m)}{\frac{1}{N}\sum_{n=1}^{N}\left|d[n]\right|^2}\right), \tag{2.80}$$

$d[n]$ being the measured output signal and m denoting a certain model. For model-structure selection, the penalised mean-squared error (2.74) with the two criteria (2.75) and (2.76), is used.

Modelling a Three-Stage High-Power LDMOS EDGE Amplifier

Fig. 2.6 shows the measurement setup for gathering input/output data. The test-signal $x[n]$ is generated in the PC. It is a multi-tone signal with a bandwidth of 1 MHz and 101 equally spaced tones of identical amplitude and with random phases. The complex digital baseband signal is then converted to two analogue signals, the in-phase component $x_I(t)$ and the quadrature-phase component $x_Q(t)$, using the Rohde&Schwarz [60] I/Q-Modulation Generator AMIQ. The analogue signals are then used to modulate a carrier at 1,9 GHz using the Rohde&Schwarz Vector Signal Generator SMIQ. A single-stage driver amplifier (Mini-Circuits ZHL-42W [58]) with a minimum gain of 30 dB boosts the signal to a sufficiently high power level for the LDMOS-EDGE high-power amplifier. After attenuation, the output signal is down-converted and demodulated with a PSA signal analyzer from Agilent Techn. [61] which delivers the complex valued baseband output signal $d[n]$ to the PC. The analysis bandwidth of the signal analyzer is limited to 8 MHz – therefore, with an input-signal bandwidth of 1 MHz, out-of-band harmonics up to the seventh order can be observed. The sampling rate of the signal analyzer is 10,24 MHz.

Tab. 2.1 shows the normalised mean-squared errors, defined in (2.80) and (2.74) for different Volterra models of the power amplifier chain of Fig. 2.6. The penalised mean-squared error

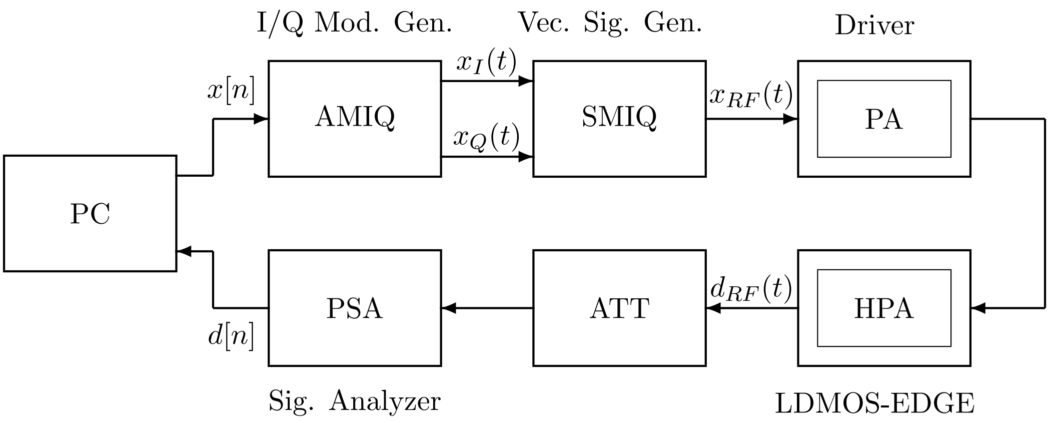


Figure 2.6: Measurement setup for input/output data generation

is normalised with the penalised mean-squared error of the model with the lowest complexity. First, the order p is increased stepwise – the specification of the Volterra model is defined by the vector $m_V = [N_1, N_3, N_5, N_7, N_9]$, N_p denoting the length of the kernel $h_p[n_1, \ldots, n_p]$ in each dimension.

m_V	order	$J_r(m)[dB]$	$W_{r,\mathrm{FPE}}(m)[dB]$	$W_{r,\mathrm{MDL}}(m)$
[1,1]	3	-33,04	0	0
[1,1,1]	5	-34,5	-1,85	-1,81
[1,1,1,1]	7	-35,07	-2,33	-2,26
[1,1,1,1,1]	9	-35,1	-2,34	-2,23

Table 2.1: Normalised mean-squared errors of static Volterra models – nonlinear order selection

From Tab. 2.1 it can be seen that a seventh order model is sufficient – however, the mean-squared error does not increase significantly even if only a fifth order model is selected.

Now the length of the linear kernel is increased stepwise in the seventh order model. Again, the penalised mean-squared errors are normalised with the lowest-complex model.

m_V	order	$J_r(m)[dB]$	$W_{r,\mathrm{FPE}}(m)[dB]$	$W_{r,\mathrm{MDL}}(m)$
[1,1,1,1]	7	-35,07	0	0
[2,1,1,1]	7	-35,43	-0,54	-0,51
[3,1,1,1]	7	-35,45	-0,58	-0,51
[4,1,1,1]	7	-35,46	-0,58	-0,48

Table 2.2: Normalised mean-squared errors of Volterra models of order seven with memory only in the linear kernel

From Tab. 2.2 it can be seen that two taps for the linear kernel are sufficient, adding more taps reduces the error only marginally, if the memory-length is increased to four, W_{MDL} begins to increase, indicating an overfit. For the estimation of the length of the third-order kernel the procedure is repeated, with the reference-model $m_V = [2, 1, 1, 1]$. Adding only one tap is also here enough. Adding taps in the higher-order kernels does not improve the estimation. The number of parameters increases drastically and the estimation becomes very sensitive due to the ill-conditioned regression matrix.

m_V	order	$J_r[dB]$	$J_{r,\text{Mag.}}[dB]$	$J_{r,\text{Phase}}[dB]$	num. of param.
Volterra - [2,2,1,1]	7	-35,7	-40,1	-41,3	10

Table 2.3: Normalised mean-squared errors of best Volterra model with memory in the linear and third order kernel

In Tab. 2.3 the performance figures of the best Volterra model for this specific measurement are reported.Here, the normalised mean-squared errors of the magnitude and phase

$$J_{r,\text{Mag.}}[dB] = 20\log\left(\frac{\Big\| |y[n]| - |d[n]| \Big\|_2}{\big\|d[n]\big\|_2}\right) \tag{2.81}$$

$$J_{r,\text{Phase}}[dB] = 20\log\left(\frac{\Big\| \arg\big(y[n]\big) - \arg\big(d[n]\big) \Big\|_2}{\Big\|\arg\big(d[n]\big)\Big\|_2}\right) \tag{2.82}$$

are reported, too.

In Tab. 2.4 the modelling results of the other models (Wiener-, Hammerstein-, and Saleh-model) are listed. The memory-lengths of the linear filters in the Wiener- and Hammerstein-model are set to two taps. The order of the non-linearity, expressed as a Taylor-series, is seven. If represented by an equivalent Volterra-series, all kernels (up to the seventh order) have a two-tap memory.

m	order	$J_r[dB]$	$J_{r,\text{Mag.}}[dB]$	$J_{r,\text{Phase}}[dB]$	num. of param.
[2,2,1,1]	7	-35,7	-40,1	-41,3	10
Wiener	7	-35,6	-39,75	-41,9	6
Hammerstein	7	-35,6	-39,78	-42,1	6
Saleh	-	-25,9	-33,3	-31,5	4(real-valued)

Table 2.4: Comparison of different models with and without memory

The Wiener-and Hammerstein models achieve about equivalent modelling results than the Volterra model using fewer parameters, the Wiener- and Hammerstein models being more accurate modelling the phase response. The estimation is also more robust – the least-squares matrices for Wiener-and Hammerstein systems (2.45), (2.48), (2.55), and (2.58) have condition numbers which are much smaller (in the range of 10) than in the case of the Volterra model (up to several hundred, depending on the model structure). The often used Saleh model (originally derived for modelling travelling-wave tube amplifiers [9]) shows the worst modelling results. Both, amplitude and phase show errors which are approx. 7-9 dB larger than the errors achieved with a Volterra-, Wiener-, or Hammerstein model.

Fig. 2.7 shows the spectra of the measured output signal and the output signal of the Wiener model with a two tap linear filter and a static non-linearity, modelled as a power series of order seven. The relative error is the power spectrum of the error $e[n] = d[n] - y[n]$, relative to the power spectrum of the measured system output signal $d[n]$

$$S_e(f) = 10\log\left(\frac{S_d(f) - S_y(f)}{S_d(f)}\right) \text{ in dB }. \tag{2.83}$$

The in-band signal in the frequency band $f/f_s \in [-0,048; 0,048]$ and the signal in the frequency bands $f/f_s \in [-0,145; -0,048]$ and $f/f_s \in [0,048; 0,145]$ are estimated accurately. Outside this region the error increases and the estimation is rather poor – the signal power in this region is more than 60 dB less than in the in-band zone which explains the poor estimation.

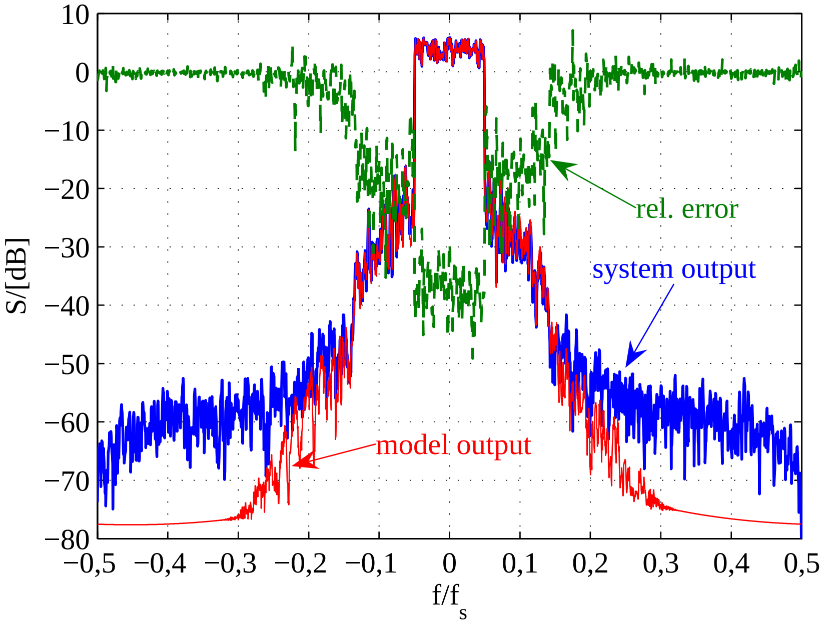


Figure 2.7: Spectra of an LDMOS EDGE power amplifier modelled with a Wiener model with a two tap linear filter and a seventh-order non-linearity – $f_s = 10,24$ MHz. The bandwidth of the input signal is 1 MHz, the analysis bandwidth is 8 MHz.

Modelling a Single-Stage Medium-Power Amplifier

A single stage power amplifier with a maximum output power (1 dB compression point) of 30 dBm and a frequency range extending over 6 GHz from 2 GHz-8 GHz (Mini-Circuits ZVE-8G [58]) is modelled next. The measurement setup differs from that in Fig. 2.6 – the setup is shown in Fig. 2.8. It uses a mixer-stage which up-converts a bandpass signal at 70 MHz, generated with a fast digital-to-analogue converter (DAC) (SUNDANCE® SMT370-module [62]), to the ISM[8]-band at 2,45 GHz. The down-converter converts the signal at 2,45 GHz to a bandpass signal at 70 MHz [63]. This signal is sampled with an analogue-to-digital converter (ADC) with sampling frequency of 100 MHz and a resolution of 14 bit (SUNDANCE® SMT370-module), producing a digital bandpass signal at 30 MHz. The signal is then stored in a fast memory-module (MEM) (SUNDANCE® SMT351-G-module [62]). The bandwidth of the analogue stages is approx. 20 MHz. The bandwidth of the input signal is 5 MHz, again it is a multi-tone signal with 101 tones, equally spaced and with random phases. The crest-factor is 5,4 dB.

---

[8] The industrial, scientific, and medical (ISM) radio bands (900 MHz, 2,4 GHz, 5,2 GHz) were originally reserved internationally for non-commercial use of RF electromagnetic fields for industrial, scientific and medical purposes. In recent years they have also been used for license-free error-tolerant communications applications such as wireless LANs and Bluetooth.

The variable attenuator between the driver power amplifier and the medium-power amplifier is used to control the saturation level of the following power amplifier.

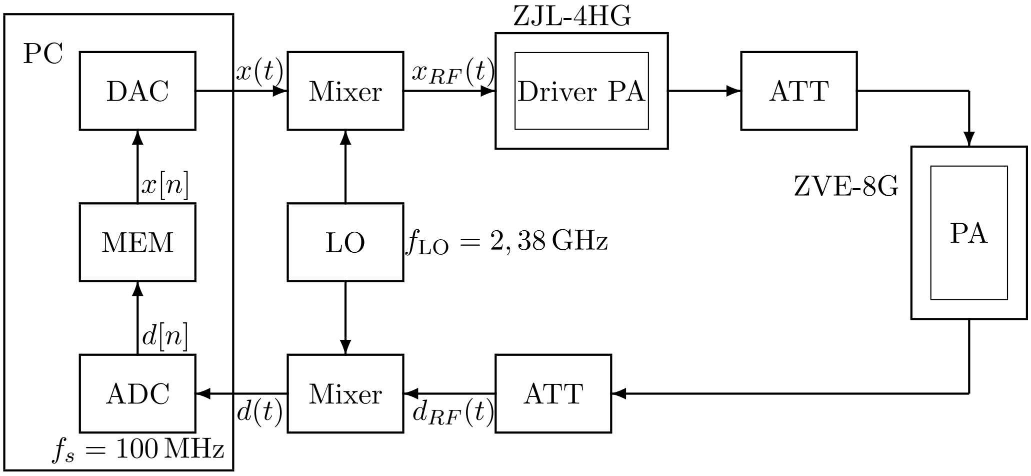


Figure 2.8: Measurement setup for input/output data generation

Tab. 2.5 shows the achieved minimal errors for different models for the Mini-Circuits medium-power amplifier ZVE-8G. The achieved errors are in general not as low as in the previous measurements – no high-level and highly accurate measurement equipment has been used. The parameter extraction is affected by the dynamic limitations of the equipment.

m	$J_r[dB]$	$J_{r,\mathrm{Mag.}}[dB]$	$J_{r,\mathrm{Phase}}[dB]$	num. of param.
Volterra - [2,1]	-22,1	-36,6	-27,1	3
Wiener	-19,5	-37,2	-24,9	4
Hammerstein	-19,6	-38,5	-24,8	4

Table 2.5: Comparison of different models with memory for the Mini-Circuits medium-power amplifier ZVE-8G

Here, the Volterra model, which is also the model with the lowest complexity, seems to be the best choice. Again, low complex models such as Wiener-or Hammerstein models, both with a two-taps linear filter and a third order power series, model the system behaviour also quite well, if compared with the general Volterra model. Memory effects are also in this case not very pronounced – due to the large frequency range of the power amplifier of 6 GHz compared with an excitation with a multi-tone signal with a bandwidth of only 5 MHz, this has to be expected. The AM-AM conversion is modelled accurately – very small errors are achieved. The AM-PM conversion is not predicted very accurately.

Fig. 2.9 shows the spectra of the system output, the model output and the relative error (2.83). The model in the figure is the Volterra model of Tab. 2.5. The model-quality degrades with increasing distance to the centre-frequency – the limited dynamic range of the equipment limits the resolution of the gathered system output-data and thus the achievable model quality.

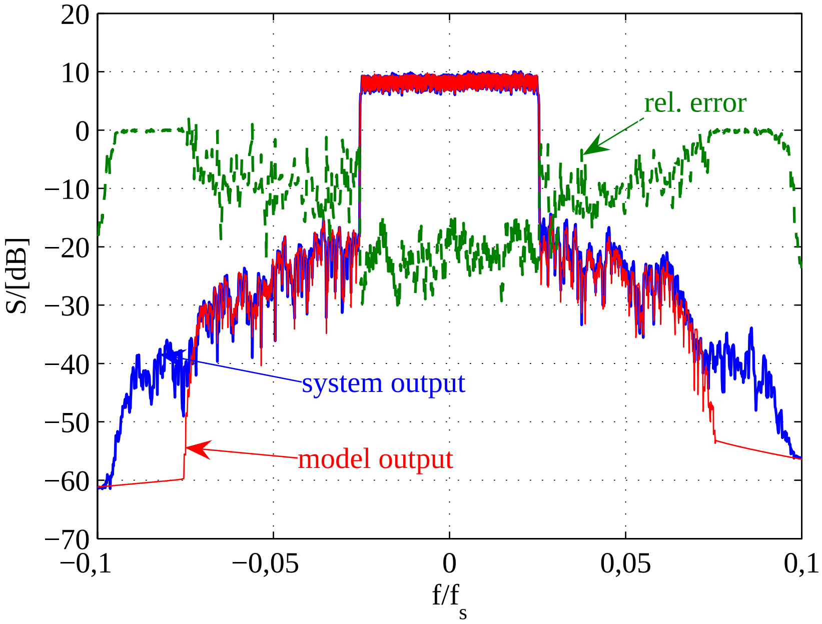


Figure 2.9: Spectra of the modelled Mini-Circuits medium-power amplifier using a third-order Volterra model. The linear kernel has two taps, the third-order kernel is memoryless ($f_s = 100\,\text{MHz}$, bandwidth of the input signal is 5 MHz, the analysis bandwidth is 20 MHz).

Modelling a Simulated WCDMA Power Amplifier

A Motorola LDMOS high power amplifier (MRF21125) for application in WCDMA is considered next. The input/output data is obtained by a simulation environment (ADS2003C [46]), which uses an equivalent-circuit based model for this power amplifier and a WCDMA input signal. Using a simulation tool for the generation of input/output data is useful to gain insight in the capabilities of the different models. Detrimental effects with negative influence on the modelling, such as measurement noise, are excluded. At the other hand, using only simulated data for the modelling can lead to a very simplistic perception of reality where non-perfect equipment has to be used.

Tab. 2.6 shows a comparison of the best models found for this data. The model structure is selected based on Rissanen's MDL-criterion (2.76). First, the nonlinear order is estimated by increasing the order of a static Volterra model step-by-step. A fifth-order model is found to be sufficient – a further increase of the model-order does not improve the estimation significantly. Then, the lengths of the kernels are increased step-by-step, observing the achieved errors (2.74) and (2.76). A length of two taps for all kernels is optimal.

The Wiener- and Hammerstein models both have a two-tap linear filter with a static nonlinearity (Taylor series) of order five. It has to be noted that the estimation of the Volterra model and Hammerstein model is very sensitive due to the ill-conditioned matrices (2.33) and (2.55) – in the case of the Volterra model the condition number is larger than 8.000, in the case of the Hammerstein model the condition number of the matrix $\mathbf{H}_{\boldsymbol{\theta}}$ is in the range of 6.000. For the Wiener model, the condition numbers of the matrices (2.45) and (2.48) are significantly smaller – in the range of 40 – which makes the estimation more robust and indicates also a good match between model and system.

m	$J_r[dB]$	$J_{r,\text{Mag.}}[dB]$	$J_{r,\text{Phase}}[dB]$	num. of param.
[2,2,2]	-53,5	-61,7	-57,1	20
Wiener	-53,5	-59,1	-56,7	5
Hammerstein	-53,4	-59,5	-54,4	5
Saleh	-17,3	-42,9	-22,1	4(real-valued)

Table 2.6: Comparison of different models for input/output data obtained with an ADS simulation of a Motorola LDMOS high-power amplifier

Fig. 2.10 shows the power spectra of the output signal of the "system" (simulation) and the Wiener model. The sampling rate is the eight-fold chip rate of 3,84 MHz which gives 30,72 MHz. The signals are normalised with the maximum value of the magnitude of the system output. The in-band signal is modelled very accurately, the accuracy of the modelling of the out-of-band components decreases with increasing distance to the in-band, recognised in the increase of the relative error, see (2.83) – most of the energy is in the in-band region, allowing therefore a very accurate modelling.

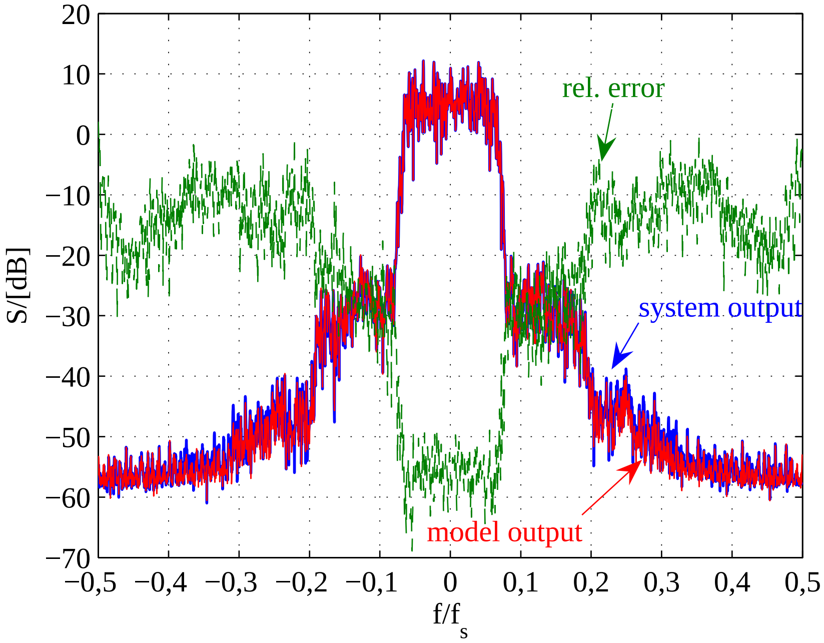
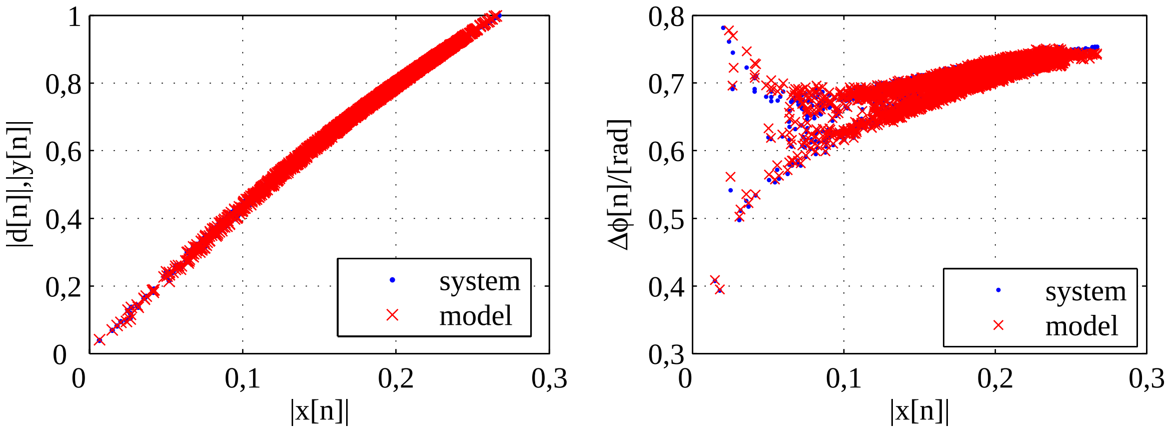


Figure 2.10: Spectra of a simulated Motorola power amplifier and the Wiener model – $f_s = 8 \cdot 3,84\,\text{MHz} = 30,72\,\text{MHz}$

In Fig. 2.11 the modelling results regarding the AM-AM and AM-PM conversion is graphically shown. The accuracy of the obtained model can be recognised immediately.

2.7 Discussion

Modelling of the different power amplifiers, using measured and simulated data, where the measurements are performed under different side-constraints – using a calibrated high-level



(a) AM-AM conversion (b) AM-PM conversion

Figure 2.11: Modelled AM-AM and AM-PM characteristics of a Motorola LDMOS power amplifier

measurement equipment in Section 2.6.1 and a non-perfect in-house [63] build up- and down-conversion stage, together with a broadband sampling system – showed that

- relatively low-order non-linearities (up to order seven) are sufficient,
- the memory effects are concentrated – kernels with two to three taps are sufficient,
- low-complex models, such as Wiener-and Hammerstein models, are adequate for all amplifiers, Volterra models do not yield significantly better results.

The necessary degree of the non-linearity depends on the saturation level of the power amplifier. Driving the amplifier in deep saturation may be efficient in terms of power-added efficiency, but the gain of the linearisation, compared to a simple input back-off, decreases, see Chapter 4 for more details. It is therefore not reasonable to use amplifiers driven in deep saturation, which would require a higher degree of non-linearity in the models.

In the considered cases the memory effects are not very pronounced – using other power amplifiers could enforce longer kernels in the models and the usage of more complex Volterra models. The memory effects are naturally associated with the signal bandwidth – in the performed measurements the bandwidth of the equipment imposes limits. With very high bandwidths, e.g., multiple-channel arrangements in WCDMA or the WLAN standard 802.11n with signal bandwidths up to 40 MHz, memory effects are expected to become important.

Chapter 3

Adaptive Identification

In this chapter low complex stochastic-gradient algorithms for the adaptive identification of the parameters of the Volterra and Wiener models presented in Chapter 2 are developed. A deterministic robustness analysis of an adaptive gradient algorithm for the parameter identification of a Volterra system and Wiener system is presented. The Wiener model requires significantly less parameters than a general Volterra model and is thus very attractive. The main disadvantage of this model consists of the fact that the description is nonlinear with respect to a part of the model-parameters, namely the linear filter parameter. The consequence is that a quadratic cost-function, which is minimised by the adaptive algorithm during the search of the optimal model-parameters, may be non-convex. This can result in a poor estimation quality if the adaptive gradient algorithm locates only a local minimum of the cost-function. Whether the algorithm finds only a local minimum or the global minimum of the cost-function depends on the initial values for the parameters which are used to start the iterative search.

Here, a two step procedure for the identification of a Wiener system is proposed and analysed in a deterministic context [15, 16]. A stochastic analysis of such an adaptive gradient identification method for a Wiener system is in practice rarely possible. The signal characteristics are most often not known and even if the signal characteristics are known, further information regarding the system, especially some knowledge about the static nonlinear function, must be available. Even if all this is known, a very restrictive convergence analysis giving only information about convergence-in-the mean assuming a small step-size can be performed [64, 65]. The deterministic approach does not rely on such a-priori knowledge making it a very general analysis approach. On the other hand, the drawback of this method is that it provides only sufficient but not necessary conditions for convergence which may be very conservative.

This chapter begins with a short introduction of the gradient identification algorithm. The deterministic analysis method for stability is presented briefly for a general adaptive Volterra filter which is linear-in-parameters. Following this introduction the gradient identification method is applied for the identification of a Wiener system. Following an idea of [66], a two-step procedure is proposed: First, an estimation of the parameters which enter nonlinearly in the description, namely the parameters of the linear filter, based on a gradient identification, is performed. A deterministic robustness analysis [13, 67] of this identification algorithm is performed [16]. In the next step, an identification algorithm for the estimation of the parameters of the static nonlinear output-filter is presented and analysed.

3.1 The Stochastic-Gradient Algorithm

Stochastic-gradient algorithms are widely used as a tool for model-parameter identification. Linear-in-parameter models lead to various variants, e.g., least-mean-squares (LMS) algorithm and normalised LMS (NLMS) algorithm. All these algorithms are obtained from the steepest-decent method by replacing the required gradient vectors with different approximations [68, 14]. By replacing these deterministic quantities by approximations, the update directions become subject to stochastic fluctuations (gradient noise).

Using these approximations for the gradient there is no need to know the signal statistics (e.g., first or second order moments) for the implementation of the algorithm. These statistical quantities are furthermore rarely available in practice. For a statistical convergence analysis these quantities must be known. Consequently, if the signal statistics are not known a stochastic analysis of these algorithms is not possible. A deterministic analysis requires far less a-priori knowledge and is thus, at least in principle, often possible, see Section 3.1.1 and Section 3.1.2 further ahead.

The stochastic gradient algorithms possess a learning mechanism which enables them to estimate the required signal statistics. Further, stochastic-gradient algorithms are able to track the signal statistics and thus system variations. They inherently possess a tracking capability. Their learning and tracking capability, together with a low-complex and robust implementation, make these algorithms very attractive for practical applications.

3.1.1 Stochastic-Gradient Identification for Linear-in-Parameter Models

Volterra models are linear-in-parameters. The input-output relation can be stated as an inner vector product

$$y[n] = \mathbf{x}_n \boldsymbol{\theta} \ , \tag{3.1}$$

where $\mathbf{x}_n$ is the (row) input-signal vector

$$\mathbf{x}_n = \left[\mathbf{x}_{1,n}, \mathbf{x}_{3,n}, \ldots, \mathbf{x}_{2P-1,n}\right] \ , \tag{3.2}$$

composed of all necessary parts corresponding to the specific Volterra model of nonlinear order $2P-1$, e.g.,

$$\begin{aligned} \mathbf{x}_{1,n} &= [x[n], x[n-1], \ldots, x[n-N_1+1]] && (3.3)\\ \mathbf{x}_{3,n} &= \Big[x[n]\big|x[n]\big|^2, x^2[n]x^*[n-1], \ldots, x[n-N_3+1]\big|x[n-N_3+1]\big|^2\Big] \ . && (3.4) \end{aligned}$$

Here, $\mathbf{x}_{1,n}$ denotes the signal-vector for the linear part and $\mathbf{x}_{3,n}$ is the signal-vector for the third-order part, N_1 is the length of the linear kernel, N_3 is the length of the third-order kernel (assumed to be equal in all three dimensions), cf. also Section 2.1.5. The corresponding filter-parameter (column-) vector comprises of

$$\boldsymbol{\theta} = \left[\boldsymbol{\theta}_1^T, \boldsymbol{\theta}_3^T, \ldots, \boldsymbol{\theta}_{2P-1}^T\right]^T \ , \tag{3.5}$$

with the linear and third order parts

$$\begin{aligned} \boldsymbol{\theta}_1 &= \left[h_1[0], h_1[1], \ldots, h_1[N_1-1]\right]^T && (3.6)\\ \boldsymbol{\theta}_3 &= \left[h_3[0,0,0], h_3[0,0,1], \ldots, h_3[N_3-1, N_3-1, N_3-1]\right]^T \ . && (3.7) \end{aligned}$$

If the observed output-signal from the system is denoted by $d[n]$, the optimal weight-vector $\boldsymbol{\theta}_W$ minimising the mean-squared error

$$J = \mathbb{E}\left\{\left|d[n] - \mathbf{x}_n\boldsymbol{\theta}\right|^2\right\} , \tag{3.8}$$

where $\mathbb{E}\{\cdot\}$ denotes the expectation operator, is the Wiener-solution, (cf., e.g., [14])

$$\boldsymbol{\theta}_W = \mathbf{R}_{xx}^{-1}\mathbf{r}_{dx} . \tag{3.9}$$

Here,

$$\begin{aligned} \mathbf{R}_{xx} &= \mathbb{E}\left\{\mathbf{x}_n^H\mathbf{x}_n\right\} && (3.10) \\ \mathbf{r}_{dx} &= \mathbb{E}\left\{d[n]\mathbf{x}_n^H\right\} , && (3.11) \end{aligned}$$

$\mathbf{R}_{xx}$ is the auto-correlation matrix of the input signal and $\mathbf{r}_{dx}$ is the cross-correlation vector between the output sample $d[n]$ and input signal-vector $\mathbf{x}_n$. The optimal weight vector results in a minimum mean-square error

$$J_W = \sigma_d^2 - \mathbf{r}_{xd}\mathbf{R}_{xx}^{-1}\mathbf{r}_{dx} \tag{3.12}$$

with $\sigma_d^2 = \mathbb{E}\left\{\left|d[n]\right|^2\right\}$.

The difficulty is the computation of the correlation matrix $\mathbf{R}_{xx}$, resp. the correlation vector $\mathbf{r}_{dx}$. Even if the statistics of the signals $x[n]$ and $d[n]$ are known, due to the complicated structure of the signal-vector $\mathbf{x}_n$, which comprises various products, see, e.g., (3.4), the computation of the expectation is in practice not feasible. Therefore, a direct solution (3.9) or a steepest-decent approach, which does not require the inversion of the correlation matrix,

$$\boldsymbol{\theta}_i = \boldsymbol{\theta}_{i-1} + \mu\left[\mathbf{r}_{dx} - \mathbf{R}_{xx}\boldsymbol{\theta}_{i-1}\right] , \ i \geq 0 , \ \boldsymbol{\theta}_{-1} = \text{initial guess} \tag{3.13}$$

with a (small) step-size μ are not feasible. Further, in practice the statistics of the signals and thus the quantities $\mathbf{R}_{xx}$ and $\mathbf{r}_{dx}$ tend to vary with time, e.g., due to a system change (in this case the power amplifier). Therefore, the optimal model-parameters $\boldsymbol{\theta}_W$ will also change. A steepest-decent approach (3.13) or a direct solution (3.9) cannot track system changes.

Stochastic-gradient algorithms provide approximations to the steepest-decent method by introducing approximations to the non-computable quantities $\mathbf{R}_{xx}$ and $\mathbf{r}_{dx}$. Further, these algorithms possess a tracking capability which enables them to track signal and/or system changes. The most popular and widely used algorithm is the least-mean-squares (LMS) algorithm. Here, the deterministic quantities $\mathbf{R}_{xx}$ and $\mathbf{r}_{dx}$ are approximated with the instantaneous estimates

$$\begin{aligned} \mathbf{R}_{xx} &\approx \mathbf{x}_n^H\mathbf{x}_n && (3.14) \\ \mathbf{r}_{dx} &\approx d[n]\mathbf{x}_n^H && (3.15) \end{aligned}$$

resulting in the LMS-algorithm

$$\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + \mu[n]\mathbf{x}_n^H\left(d[n] - \mathbf{x}_n\boldsymbol{\theta}_{n-1}\right) , \ n \geq 0 , \ \boldsymbol{\theta}_{-1} = \text{initial guess} . \tag{3.16}$$

The iteration index i used in the steepest-descent method has been changed to the time index n – with each new data sample $d[n]$ and associated signal vector $\mathbf{x}_n$ a new estimate for the parameter-vector $\boldsymbol{\theta}$ is formed by adding a specific correction term to the old estimate $\boldsymbol{\theta}_{n-1}$. The fixed step-size μ has also been changed to a time-variant step-size $\mu[n]$. Various selections for the step-size exist, giving rise to several variants of the classical LMS-algorithm. In Tab. 3.1 the most common modifications for the step-size are listed, together with the name of the algorithm.

Algorithm	step-size
LMS, constant step-size	μ
LMS, time-variant step-size	$\mu[n]$
normalised LMS (NLMS)	$\alpha/\|\mathbf{x}_n\|_2^2$
ε-NLMS	$\alpha/\left(\varepsilon+\|\mathbf{x}_n\|_2^2\right)$
a-posteriori form LMS	$\alpha/\left(1+\alpha\|\mathbf{x}_n\|_2^2\right)$

Table 3.1: List of common stochastic-gradient algorithms with corresponding step-sizes

Convergence Analysis – Error-Vector in the Mean

For a statistical convergence analysis the error-vector form of the stochastic-gradient algorithm (3.16) is used,

$$\tilde{\boldsymbol{\theta}}_n = \tilde{\boldsymbol{\theta}}_{n-1} - \mu[n]\mathbf{x}_n^H \tilde{e}_a[n] \;, \tag{3.17}$$

where $\tilde{\boldsymbol{\theta}}_n = \boldsymbol{\theta}_* - \boldsymbol{\theta}_n$ denotes the parameter error-vector and

$$\tilde{e}_a[n] = d[n] - \mathbf{x}_n\boldsymbol{\theta}_{n-1} \tag{3.18}$$

is the disturbed a-priori error. If a reference model is assumed so that

$$d[n] = \mathbf{x}_n\boldsymbol{\theta}_* + v[n] \;, \tag{3.19}$$

$v[n]$ denoting zero-mean white noise, independent from $\mathbf{x}_n$, the error-vector can be written as

$$\tilde{\boldsymbol{\theta}}_n = \left(\mathbf{I} - \mu[n]\mathbf{x}_n^H\mathbf{x}_n\right)\tilde{\boldsymbol{\theta}}_{n-1} - \mu[n]\mathbf{x}_n^H v[n] \;. \tag{3.20}$$

Taking the expectation on both sides (assuming further statistical independence of $\mathbf{x}_n$ and $\tilde{\boldsymbol{\theta}}_{n-1}$) gives

$$\mathbb{E}\left\{\tilde{\boldsymbol{\theta}}_n\right\} = \left(\mathbf{I} - \mu[n]\mathbf{R}_{xx}\right)\mathbb{E}\{\tilde{\boldsymbol{\theta}}_{n-1}\} \;. \tag{3.21}$$

The necessary condition for convergence in the mean is (as in the case of the steepest-descent method)

$$0 < \mu[n] < \frac{2}{\lambda_{\max}} \;, \tag{3.22}$$

where $\lambda_{\max} = \max\bigl(\lambda(\mathbf{R}_{xx})\bigr)$, $\lambda(\mathbf{R}_{xx})$ denoting the spectrum of the correlation-matrix $\mathbf{R}_{xx}$. It is necessary to have knowledge of the signal statistic, especially the high-order moments of the input-signal $x[n]$ are required. In contrast to the linear filter case, even for a convergence analysis considering only the mean of the error-vector, moments of higher than second order are required. As an example, (3.23) presents the correlation matrix for a very simple third order Volterra filter $m_V = [N_1, N_3]$ with $N_1 = 2$, corresponding to a two-tap linear part and $N_3 = 1$, corresponding to a memoryless third-order part,

$$\mathbf{R}_{xx} = \begin{pmatrix} \mathbb{E}\{|x[n]|^2\} & \mathbb{E}\{x^*[n]x[n-1]\} & \mathbb{E}\{|x[n]|^4\} \\ \mathbb{E}\{x[n]x^*[n-1]\} & \mathbb{E}\{|x[n-1]|^2\} & \mathbb{E}\{x[n]x^*[n-1]|x[n]|^2\} \\ \mathbb{E}\{|x[n]|^4\} & \mathbb{E}\{x^*[n]x[n-1]|x[n]|^2\} & \mathbb{E}\{|x[n]|^6\} \end{pmatrix} . \tag{3.23}$$

Moments up to the sixth-order are required. For a Volterra system of order P, moments up to the order $2P$ are required for this convergence analysis. In most cases it is not feasible to compute these moments. Thus, it is not possible to compute the upper bound for the step-size as given in (3.22). Since in general no upper bound for the step-size can be computed, small step-sizes are used in order to guarantee convergence.

Furthermore, the terms involving products of the input signal in the correlation matrix have the negative effect to enlarge the eigenvalue spread of the correlation matrix significantly. If, e.g., $x[n] \sim \mathcal{B}(\pm 1)$, i.e., $x[n]$ takes on values $+1$ or -1 with equal probability, and the process is white, the correlation matrix $\mathbf{R}_{xx}$ becomes simply

$$\mathbf{R}_{xx} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} . \tag{3.24}$$

This matrix is singular, thus having an eigenvalue spread which is infinity.

Since the convergence-speed is small if the eigenvalue spread is large, the convergence-speed of nonlinear adaptive Volterra filters can in general be expected to be small compared to adaptive linear filters.

Deterministic Robustness Analysis – Local Passivity

The difficulty in computing the high-order moments prevents a statistical analysis (convergence in-the-mean, mean-square behaviour) of the stochastic-gradient algorithm (3.16) for Volterra filters in most cases. A deterministic approach involving energy relations provides a method to analyse whether the gradient-algorithm (3.16) is guaranteed to converge [13] to a final estimate for the optimal parameter-vector $\boldsymbol{\theta}_*$ or not.

Rewriting the error-vector update-equation (3.17)

$$\tilde{\boldsymbol{\theta}}_n = \tilde{\boldsymbol{\theta}}_{n-1} - \mu[n]\mathbf{x}_n^H \left(e_a[n] + v[n]\right) \tag{3.25}$$

with the undisturbed a-priori error $e_a[n] = \mathbf{x}_n\boldsymbol{\theta}_{n-1}$ and taking the squared ℓ_2-norm on both sides gives, after rearranging terms,

$$\|\tilde{\boldsymbol{\theta}}_n\|_2^2 - \|\tilde{\boldsymbol{\theta}}_{n-1}\|_2^2 + \mu[n]\big|e_a[n]\big|^2 - \mu[n]\big|v[n]\big|^2 = \mu[n]\big|e_a[n] + v[n]\big|^2\big(\mu[n]\|\mathbf{x}_n\|_2^2 - 1\big) . \tag{3.26}$$

Whether the right-hand side is positive, negative or zero depends on the quantity $\mu[n]\|\mathbf{x}_n\|_2^2 - 1$. Compactly, the energy relation can be written as [13]

$$\frac{\|\tilde{\boldsymbol{\theta}}_n\|_2^2 + \mu[n]\big|e_a[n]\big|^2}{\|\tilde{\boldsymbol{\theta}}_{n-1}\|_2^2 + \mu[n]\big|v[n]\big|^2} \begin{cases} < 1, & \text{if} \quad \mu[n]\|\mathbf{x}_n\|_2^2 < 1 \\ = 1, & \text{if} \quad \mu[n]\|\mathbf{x}_n\|_2^2 = 1 \\ > 1, & \text{if} \quad \mu[n]\|\mathbf{x}_n\|_2^2 > 1 \end{cases} . \tag{3.27}$$

The first inequality guarantees that, no matter what the value of $v[n]$ is and no matter how far away the estimate $\boldsymbol{\theta}_{n-1}$ from the optimal value $\boldsymbol{\theta}_*$ is, it holds that

$$\|\tilde{\boldsymbol{\theta}}_n\|_2^2 + \mu[n]\big|e_a[n]\big|^2 < \|\tilde{\boldsymbol{\theta}}_{n-1}\|_2^2 + \mu[n]\big|v[n]\big|^2 . \tag{3.28}$$

The map $\mathbb{T}_n$, mapping the signals $\left\{\tilde{\boldsymbol{\theta}}_{n-1}, \sqrt{\mu[n]}v[n]\right\}$ to the output signals $\left\{\tilde{\boldsymbol{\theta}}_n, \sqrt{\mu[n]}e_a[n]\right\}$ is therefore passive, i.e.,

$$\begin{bmatrix} \tilde{\boldsymbol{\theta}}_n \\ \sqrt{\mu[n]}e_a[n] \end{bmatrix} = \mathbb{T}_n \begin{bmatrix} \tilde{\boldsymbol{\theta}}_{n-1} \\ \sqrt{\mu[n]}v[n] \end{bmatrix} , \text{ with } \|\mathbb{T}_n\|_2^2 < 1 . \tag{3.29}$$

The step-size selection according to an NLMS and an ε-NLMS with $\alpha < 1$ and the a-posteriori LMS with $\alpha > 0$, cf. Tab. 3.1, guarantees local stability since the step-sizes are guaranteed to be smaller than the limit

$$\bar{\mu}[n] = \frac{1}{\|\mathbf{x}_n\|_2^2} \ . \tag{3.30}$$

Deterministic Robustness Analysis – Global Passivity

In the previous analysis, local passivity of the map $\mathbb{T}_n$, i.e., from time-index $n-1$ to n, was guaranteed if $\mu[n] < \bar{\mu}[n]$. For the stability-analysis over the finite time-horizon $n = 0, \dots, N$, the sum on both sides of (3.28) is performed

$$\sum_{n=0}^{N} \left(\|\tilde{\boldsymbol{\theta}}_n\|_2^2 + \mu[n]\big|e_a[n]\big|^2 \right) < \sum_{n=0}^{N} \left(\|\tilde{\boldsymbol{\theta}}_{n-1}\|_2^2 + \mu[n]\big|v[n]\big|^2 \right) \ , \tag{3.31}$$

resulting in

$$\|\tilde{\boldsymbol{\theta}}_N\|_2^2 + \sum_{n=0}^{N} \mu[n]\big|e_a[n]\big|^2 \leq \|\tilde{\boldsymbol{\theta}}_{-1}\|_2^2 + \sum_{n=0}^{N} \mu[n]\big|v[n]\big|^2 \ . \tag{3.32}$$

If

$$\mu[n]\|\mathbf{x}_n\|_2^2 < 1 \text{ for } 0 \leq n \leq N \ , \tag{3.33}$$

the algorithm behaves also globally passive, i.e., the error-energy $\|\tilde{\boldsymbol{\theta}}_N\|_2^2 + \sum_{n=0}^{N} \mu[n]\big|e_a[n]\big|^2$, comprising the final remaining distance from the optimal vector $\boldsymbol{\theta}_*$ and the accumulated sum of the undisturbed a-priori errors is guaranteed to be smaller than the energy of the initial weight-distance and the noise, $\|\tilde{\boldsymbol{\theta}}_{-1}\|_2^2 + \sum_{n=0}^{N} \mu[n]\big|v[n]\big|^2$.

Feedback Structure – Local and Global Passivity

Equation (3.25) can easily be transformed to

$$\tilde{\boldsymbol{\theta}}_n = \tilde{\boldsymbol{\theta}}_{n-1} - \bar{\mu}[n]\mathbf{x}_n^H \left(e_a[n] + \bar{v}[n]\right) \tag{3.34}$$

with the abbreviation

$$\bar{v}[n] = \frac{\mu[n]}{\bar{\mu}[n]} v[n] - \left(1 - \frac{\mu[n]}{\bar{\mu}[n]}\right) e_a[n] \ . \tag{3.35}$$

Since now the step-size is $\bar{\mu}[n]$, the forward map $\bar{\mathbb{T}}_n$, mapping $\{\boldsymbol{\theta}_{n-1}, \sqrt{\bar{\mu}[n]}\bar{v}[n]\}$ to the output signals $\{\boldsymbol{\theta}_n, \sqrt{\bar{\mu}[n]}e_a[n]\}$, is lossless, $\|\bar{\mathbb{T}}_n\|_2^2 = 1$. A feedback path exists, which produces the modified noise signal $\bar{v}[n]$. Together with the forward map, the system equations read

$$\begin{aligned}
\begin{bmatrix} \tilde{\boldsymbol{\theta}}_n \\ \sqrt{\bar{\mu}[n]}e_a[n] \end{bmatrix} &= \bar{\mathbb{T}}_n \begin{bmatrix} \tilde{\boldsymbol{\theta}}_{n-1} \\ \sqrt{\bar{\mu}[n]}\bar{v}[n] \end{bmatrix} \ , \text{ with } \|\bar{\mathbb{T}}_n\|_2^2 = 1 && (3.36) \\
\sqrt{\bar{\mu}[n]}\bar{v}[n] &= \frac{\mu[n]}{\bar{\mu}[n]} \sqrt{\bar{\mu}[n]}v[n] - \left(1 - \frac{\mu[n]}{\bar{\mu}[n]}\right) \sqrt{\bar{\mu}[n]}e_a[n] \ . && (3.37)
\end{aligned}$$

This feedback-system is illustrated in Fig. 3.1.

For local stability it suffices that the magnitude of the gain of the feedback loop g_{FB}, which is

$$g_{\mathrm{FB}} = 1 - \frac{\mu[n]}{\bar{\mu}[n]} \ , \tag{3.38}$$

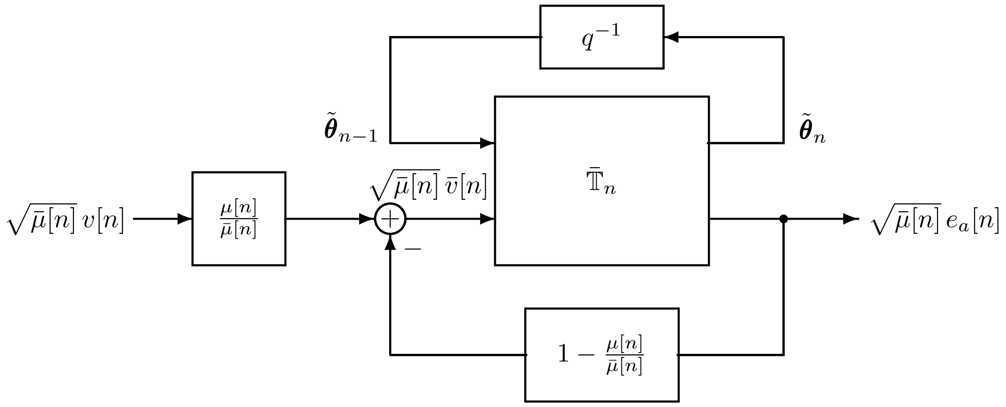

q^{-1}
$\tilde{\boldsymbol{\theta}}_{n-1}$
$\tilde{\boldsymbol{\theta}}_n$
$\bar{\mathbb{T}}_n$
$\sqrt{\bar{\mu}[n]}\, v[n]$
$\frac{\mu[n]}{\bar{\mu}[n]}$
$\sqrt{\bar{\mu}[n]}\, \bar{v}[n]$
$-$
$\sqrt{\bar{\mu}[n]}\, e_a[n]$
$1 - \frac{\mu[n]}{\bar{\mu}[n]}$

Figure 3.1: Illustration of LMS filters feedback system based on (3.36), (3.37)

is smaller than one – the forward map is lossless (small gain theorem [69]). Therefore, local stability can be assured if

$$0 < \mu[n] < 2\bar{\mu}[n] \;, \tag{3.39}$$

where the upper bound is now twice as large as the upper bound obtained in (3.33).

The global analysis for the finite time-horizon, see Appendix A.1, yields the bound for the step-size

$$0 < \mu[n] < 2\bar{\mu}[n] \;,\; n = 0, \ldots, N \;. \tag{3.40}$$

The mentioned algorithms in Tab. 3.1, especially the NLMS and ε-NLMS with $\alpha < 2$, and the a-posteriori LMS with $\alpha > 0$, fulfill the requirement for a globally stable system.

With this deterministic analysis, requiring only the computation of the signal energies, bounds for the step-sizes for stochastic-gradient algorithms for general Volterra systems can be devised.

Example – Learning Behaviour of an Adaptive Volterra Filter

For illustration of the learning-behaviour of adaptive Volterra filters a third order Volterra system is identified using the stochastic gradient algorithms listed in Tab. 3.1. The parameters of the Volterra-system are $\boldsymbol{\theta}_* = [3, 3 + j2, 5; 0, 18 + j0, 12; -0, 5 - j0, 3]^T$, the system is driven with a multitone-signal. The system is only mildly nonlinear, the coefficient of the third-order term being relatively small compared to the linear filter tap-weights. In order to work with a realistic system, the parameters are taken from a least-squares estimation using the input- and output signals of the simulated WCDMA high power amplifier, see Section 2.6.1. Fig. 3.2 shows the relative misadjustment of the gradient-algorithms from Tab. 3.1, averaged over 50 independent simulation runs,

$$m[n] = \mathbb{E}\left(\frac{\|\tilde{\boldsymbol{\theta}}_n\|_2^2}{\|\tilde{\boldsymbol{\theta}}_{-1}\|_2^2} \right) \approx \frac{1}{50} \sum_{r=1}^{50} \frac{\|\tilde{\boldsymbol{\theta}}_n(r)\|_2^2}{\|\tilde{\boldsymbol{\theta}}_{-1}\|_2^2} \;, \tag{3.41}$$

whereby r denotes the simulation run, $\tilde{\boldsymbol{\theta}}_{-1} = \boldsymbol{\theta}_* - \boldsymbol{\theta}_{-1}$ is the initial error, where the starting-guess $\boldsymbol{\theta}_{-1} = \mathbf{0}$ in all simulations. A small amount of noise, drawn from a zero-mean, white gaussian process, is added to the output of the reference system, resulting in an $\mathrm{SNR} = 80\,\mathrm{dB}$

at the output of the system. Tab. 3.2 gives an overview of the step-size parameters used for each identification algorithm.

LMS	NLMS	ε-NLMS	a-posteriori LMS
$\mu = 0,5$	$\alpha = 1$	$\alpha = 1, \varepsilon = 10^{-4}$	$\alpha = 1$

Table 3.2: Step-size parameters used in the simulation

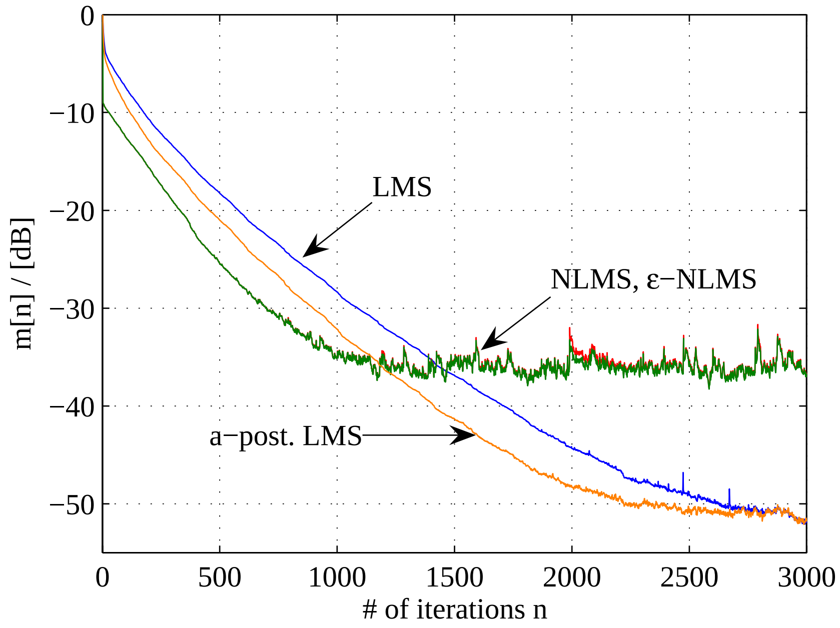
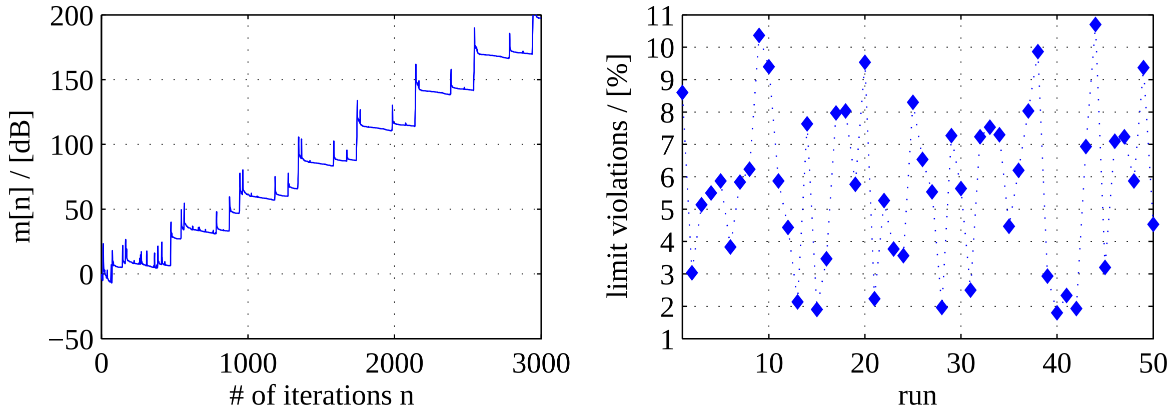


Figure 3.2: Learning curves for different stochastic-gradient algorithms identifying a third-order Volterra-system.

NLMS, ε-NLMS, and a-posteriori LMS are guaranteed to be stable in the ℓ_2-sense, since these algorithms never exceed the derived limits for the step-size (3.40). The LMS-algorithm with the fixed step-size $\mu = 0,5$ converges, the upper limit $2\bar{\mu}[n]$ is exceeded during the time-horizon $n = 0, \ldots, 3000$ in very few cases. The convergence-rate is slow, even for this minimal system with only three coefficients. The achieved minimal misadjustment is relatively poor, with LMS and a-posteriori LMS approx. -50 dB can be achieved, the a-posteriori LMS algorithm having the advantage that it is guaranteed to be stable, whereas the LMS algorithm can become unstable, even if the step-size is very small. The NLMS algorithm as well as the ε-NLMS algorithm result in a misadjustment of only -36 dB, 14 dB larger than the misadjustment achieved with the a-posteriori LMS. The former two algorithms have larger step-sizes than the a-posteriori LMS, resulting in a relatively high gradient-noise which is responsible for the poor identification quality.

If the step-size is increased to $\mu[n] = 1,5$ the LMS-algorithm becomes unstable, as can be seen from Fig. 3.3(a). Observation of $\bar{\mu}[n]$ reveals that in relatively few cases during the whole time-horizon the step-size limit $2\bar{\mu}[n]$ is exceeded. Fig. 3.3(b) shows the amount of the cases where the limit is exceeded for every simulation run. Maximally in 10-11 % of the iterations in some realisations the fixed step-size $\mu[n] = 1,5$ is larger than the limit $2\bar{\mu}[n]$. In the mean this limit is exceeded in approx. 5-6 % of the iterations. The upper-limits for the step-size which guarantee ℓ_2-stability are therefore relatively tight.



(a) Learning curve

(b) Limit violations

Figure 3.3: Unstable LMS-algorithm if the step-size is increased to $\mu = 1,5$

3.1.2 Stochastic-Gradient Identification of a Wiener System

A Wiener system is an interconnection of a linear filter $\mathbb{G}(q^{-1})$ and a static non-linearity $f(\cdot)$, see Fig. 3.4.

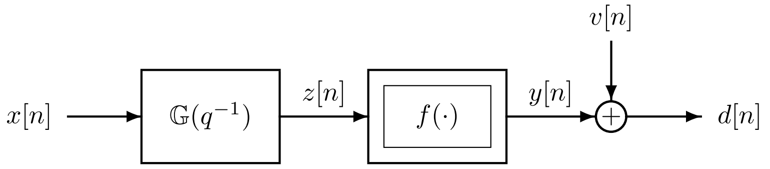


Figure 3.4: Wiener system

In this section the gradient identification of both parts of this system is investigated. The input/output relation is

$$d[n] = f\Big(\mathbb{G}\big(x[n]\big)\Big) + v[n] \ , \tag{3.42}$$

where $x[n]$ and $d[n]$ are the input and output signals, respectively. It is a simple model of a nonlinear power amplifier, experiencing both, nonlinear AM/AM and nonlinear AM/PM conversion, as well as memory effects. The noise $v[n]$ is added to account for model uncertainties and measurement noise.

If the static nonlinear function is fixed and known, only the parameters of the linear filter have to be identified. In this case the input/output description is in general nonlinear with respect to the linear filter parameters. A gradient-type algorithm, minimising a quadratic cost-function, may identify only a local minimum of the error-surface. Nevertheless, such approaches occur frequently in applications, e.g., in the neural network context [68]. A deterministic robustness analysis can be performed also in this case, see [67].

If the static non-linearity is not known a-priori, a parametrisation of the static nonlinear function is required, e.g., $f(\cdot) = \sum_{p=0}^{P-1} \theta_p \, \phi_p(\cdot)$, using a set of basis functions $\{\phi_p(\cdot)\}$, e.g., orthogonal polynomials or simply powers $\phi_p\left(z[n]\right) = z[n]\big|z[n]\big|^{2p}$ can be used. This power series with only uneven powers is considered here, cf. also the discussion in Section 2.2.

The problem is that two sets of parameters, the linear filter parameters and the parameters of the static nonlinear function, have to be identified. For the adaptive identification, a two step procedure is applied:

1. In the first step, the parameters of the linear filter are estimated.
2. In the second step, the parameters of the static nonlinear function are identified.

The accuracy of the estimation of the linear filter parameters depends on the nonlinear function. The output of the Wiener system (without noise) is

$$y[n] = \sum_{p=0}^{P-1} \theta_p \mathbb{G}\big(x[n]\big) \Big|\mathbb{G}\big(x[n]\big)\Big|^{2p} = \underbrace{\theta_0 \mathbb{G}\big(x[n]\big)}_{\text{linear part}} + \underbrace{\mathbb{G}\big(x[n]\big) \sum_{p=1}^{P-1} \theta_p \Big|\mathbb{G}\big(x[n]\big)\Big|^{2p}}_{\text{nonlinear part}} , \tag{3.43}$$

where $\mathbb{G}(q^{-1}) = \sum_{i=0}^{N-1} g_i q^{-i}$ is the linear filter, here assumed to be an FIR structure. It can be seen that the linear gain can be attributed exclusively to one of the two parts, either the linear filter or the static non-linearity [70]. E.g., the linear filter can be constrained to be monic ($g_0 = 1$) by attributing the linear gain entirely to the linear part of the static nonlinear function. Compactly, (3.43) can be written as

$$y[n] = \theta_0 z[n] + \underbrace{z[n] \sum_{p=1}^{P-1} \theta_p \big|z[n]\big|^{2p}}_{\text{nonlinear distortion}} , \tag{3.44}$$

where $z[n] = \mathbb{G}\big(x[n]\big)$, now $\mathbb{G}(\cdot)$ being monic. The nonlinear part can be seen as an additive disturbance term, which is small if the signal amplitude $\big|z[n]\big|$ is small. Consequently, a method for the estimation of the linear filter parameters could be to excite the system with a small signal amplitude and estimate the linear filter parameters only. The disadvantage of this method is that the dynamic effects of the system are in general dependent of the signal amplitude (e.g., due to heating effects). Further, it might not be possible to use a signal with small amplitude for the estimation of the linear filter parameters.

First Step: Parameter Estimation for Linear Filter

In Fig. 3.5 the adaptive scheme for the identification of the linear filter parameters is shown [16], following an idea from [66]. The nonlinear distortions produced by the static nonlinear function at the output of the linear filter, see Fig. 3.4 and (3.43), are reduced by the parameterised nonlinear function $g(\cdot)$. If the noise $v[n]$ vanishes and the nonlinear function $g(\cdot) = f^{-1}(\cdot)$, the nonlinear distortion is compensated completely and the problem reduces to the adaptive identification of a linear filter. In the general case, the inverse $f^{-1}(\cdot)$ is not known. The inverse $f^{-1}(\cdot)$ is therefore approximated using a truncated series

$$f^{-1}(\cdot) \approx g(\cdot) = \sum_{k=1}^{K} w_k \psi_k(\cdot) , \tag{3.45}$$

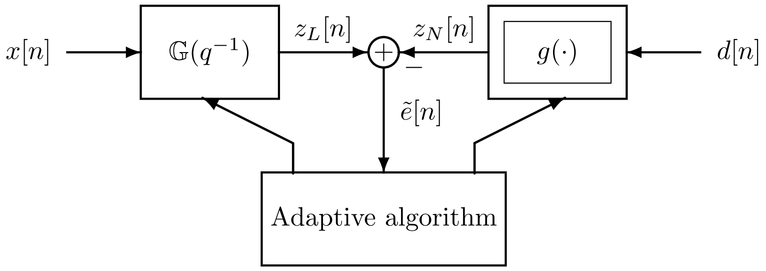


Figure 3.5: Adaptive identification of the linear filter of a Wiener system

where $\{\psi_k(\cdot)\}_{k=1}^{K}$ is a set of basis functions. For adjusting the parameters of the linear filter and the approximated inverse of the static non-linearity $f(\cdot)$, the error

$$\tilde{e}[n] = \mathbb{G}\big(x[n]\big) - g\big(d[n]\big) = z_L[n] - z_N[n] \tag{3.46}$$

is used. Both signals, $x[n]$ and $d[n]$ are accessible for the identification. The quadratic objective function

$$J[n] = \mathbb{E}\Big\{\big|\tilde{e}[n]\big|^2\Big\} \tag{3.47}$$

is defined. Since both parts, the linear filter $\mathbb{G}(\cdot)$ and the nonlinear function $g(\cdot)$ are linear-in-parameters, the objective function is convex with respect to both parameter sets. Therefore, no local minima occur. Explicitly, the objective function is

$$J[n] = \mathbb{E}\Big\{\big|z_L[n] - z_N[n]\big|^2\Big\} = \mathbb{E}\big\{|\mathbf{x}_n\mathbf{g} - \boldsymbol{\psi}_n\mathbf{w}|^2\big\} \ . \tag{3.48}$$

The parameters of the linear filter are subsumed in the (column-) vector $\mathbf{g} = [g_0, g_1, \dots, g_{N-1}]^T$, the input signal for the linear filter is a row-vector $\mathbf{x}_n = \big[x[n], x[n-1], \dots, x[n-N+1]\big]$, the parameters of the nonlinear map $g(\cdot)$ are subsumed in the column-vector $\mathbf{w} = [w_1, w_2, \dots, w_K]^T$ and the input signal for this map is $\boldsymbol{\psi}_n = \big[\psi_1\big(d[n]\big), \psi_2\big(d[n]\big), \dots, \psi_K\big(d[n]\big)\big]$. The trivial solution $\mathbf{g} = \mathbf{w} = \mathbf{0}$ has to be excluded. This can be achieved by assuming a monic linear filter, therefore fixing $g_0 = 1$. The linear gain is entirely attributed to the static non-linearity $f(\cdot)$. Therefore, the error is

$$\tilde{e}[n] = x[n] + \mathbf{x}_n^{'}\mathbf{g}^{'} - \boldsymbol{\psi}_n\mathbf{w} \ , \tag{3.49}$$

using the reduced vectors $\mathbf{x}_n^{'} = \big[x[n-1], x[n-2], \dots, x[n-N+1]\big]$ and $\mathbf{g}^{'} = [g_1, \dots, g_{N-1}]^T$.

The optimal solution minimising (3.48) is

$$\{\mathbf{g}_*^{'}, \mathbf{w}_*\} = \arg\min_{\mathbf{g}^{'},\mathbf{w}} J[n] \ . \tag{3.50}$$

A direct solution is not feasible since the required signal statistics are not known. Even if the statistics of the input signal $\mathbf{x}_n$ are known, the statistical properties of the output signal $d[n]$, required to determine the statistics of the signal $\boldsymbol{\psi}_n$, cannot be known since the nonlinear function $f(\cdot)$ is not known. Therefore, a stochastic-gradient algorithm is conceived which approximates the optimal solution and does not require knowledge of the signal statistics.

Derivation of the objective function with respect to the parameter vectors $\mathbf{g}^{'}$ and $\mathbf{w}$ and simplification of the expectation leads to the following adaptive gradient-algorithm:

$$\mathbf{g}_n^{'} = \mathbf{g}_{n-1}^{'} - \mu_g[n]\tilde{e}_a[n]\mathbf{x}_n^{'H} \ , \quad n \geq 0 \ , \mathbf{g}_{-1}^{'} \text{ given,} \tag{3.51}$$

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu_w[n]\tilde{e}_a[n]\boldsymbol{\psi}_n^H \ , \quad n \geq 0 \ , \mathbf{w}_{-1} \text{ given .} \tag{3.52}$$

Here,

$$\tilde{e}_a[n] = x[n] + \mathbf{x}_n^{'}\mathbf{g}_{n-1}^{'} - \boldsymbol{\psi}_n\mathbf{w}_{n-1} \tag{3.53}$$

is the disturbed a-priori error. Differently to the cases considered before, the disturbance $v[n]$ is indirectly contained in this error, i.e., via the signal $\boldsymbol{\psi}_n$.

For the subsequent analysis, the update-equations are defined in the error-vector form. With the optimal values for the parameter-vectors $\mathbf{g}_*^{'}$ and $\mathbf{w}_*$, the parameter error-vectors $\tilde{\mathbf{g}}_n^{'} = \mathbf{g}_*^{'} - \mathbf{g}_n^{'}$ and $\tilde{\mathbf{w}}_n = \mathbf{w}_* - \mathbf{w}_n$ are defined. The update equations for the error-vectors read

$$\begin{aligned} \tilde{\mathbf{g}}_n^{'} &= \tilde{\mathbf{g}}_{n-1}^{'} + \mu_g[n]\tilde{e}_a[n]\mathbf{x}_n^{'H} & (3.54)\\ \tilde{\mathbf{w}}_n &= \tilde{\mathbf{w}}_{n-1} - \mu_w[n]\tilde{e}_a[n]\boldsymbol{\psi}_n^H \;. & (3.55) \end{aligned}$$

The update equations (3.54) and (3.55) are coupled via the disturbed a-priori error. The task is to analyse the stability of the update equations and possibly device bounds for the step-sizes $\mu_w[n]$ and $\mu_g[n]$ which guarantee a stable operation of the algorithm. In the following, a deterministic convergence analysis providing bounds for the step-sizes is carried out.

Local Passivity Relations For the derivation of the local passivity relations the disturbed a-priori error is decomposed into

$$\tilde{e}_a[n] = e_{a,w}[n] - e_{a,g}[n] + v_e[n] \tag{3.56}$$

with $e_{a,w}[n] = \boldsymbol{\psi}_n\tilde{\mathbf{w}}_{n-1}$, $e_{a,g}[n] = \mathbf{x}_n^{'}\tilde{\mathbf{g}}_{n-1}^{'}$, and $v_e[n] = v_f[n] + v_v[n]$, where

$$\begin{aligned} v_f[n] &= f^{-1}\big(d[n]\big) - \mathbf{w}_*\boldsymbol{\psi}_n & (3.57)\\ v_v[n] &= f^{-1}\big(y[n]\big) - f^{-1}\big(d[n]\big) & (3.58) \end{aligned}$$

are reflecting the errors due to the approximation of $f^{-1}(\cdot)$ using $g(\cdot)$, see (3.45), and the influence of the noise $v[n]$. Assuming that $f^{-1}(\cdot)$ is analytic, it can be represented by a Taylor series. Thus, (3.58) can be simplified to

$$v_v[n] = f^{-1}\big(y[n]\big) - f^{-1}\big(y[n] + v[n]\big) = -\partial_y f^{-1}\big(y[n]\big)v[n] + \mathcal{O}\big(v[n]^2\big) \;, \tag{3.59}$$

which reveals that, assuming small disturbances, the noise $v_v[n]$ depends on the first derivative of the inverse at the point $y[n]$, $\partial_y f^{-1}\big(y[n]\big)$. Depending on this function the noise can be amplified or attenuated. This disturbance term vanishes if the noise $v[n]$ vanishes, in contrast to the disturbance term $v_f[n]$.

The disturbed a-priori error can be written as

$$\tilde{e}_a[n] = e_{a,w}[n] + v_w[n] = -\big(e_{a,g}[n] + v_g[n]\big) \tag{3.60}$$

introducing the new noise terms

$$\begin{aligned} v_w[n] &= -e_{a,g}[n] + v_e[n] & (3.61)\\ v_g[n] &= -e_{a,w}[n] - v_e[n] \;. & (3.62) \end{aligned}$$

The update equations in the error-vector form read now

$$\begin{aligned} \tilde{\mathbf{g}}_n^{'} &= \tilde{\mathbf{g}}_{n-1}^{'} - \mu_g[n]\big(e_{a,g}[n] + v_g[n]\big)\mathbf{x}_n^{'H} & (3.63)\\ \tilde{\mathbf{w}}_n &= \tilde{\mathbf{w}}_{n-1} - \mu_w[n]\big(e_{a,w}[n] + v_w[n]\big)\boldsymbol{\psi}_n^H \;. & (3.64) \end{aligned}$$

The equations are coupled via the noise terms $v_g[n]$ and $v_w[n]$, which depend on the undisturbed a-priori errors $e_{a,w}[n]$ and $e_{a,g}[n]$, respectively, see (3.61) and (3.62). The local passivity relations can now easily be devised, see Appendix A.2: If

$$0 < \mu_g[n] < \frac{1}{\|\mathbf{x}_n'\|_2^2} = \bar{\mu}_g[n] \tag{3.65}$$

$$0 < \mu_w[n] < \frac{1}{\|\boldsymbol{\psi}_n\|_2^2} = \bar{\mu}_w[n] \ , \tag{3.66}$$

then

$$\frac{\|\tilde{\mathbf{g}}_n'\|_2^2 + \mu_g[n]\big|e_{a,g}[n]\big|^2}{\|\tilde{\mathbf{g}}_{n-1}'\|_2^2 + \mu_g[n]\big|v_g[n]\big|^2} < 1 \tag{3.67}$$

$$\frac{\|\tilde{\mathbf{w}}_n\|_2^2 + \mu_w[n]\big|e_{a,w}[n]\big|^2}{\|\tilde{\mathbf{w}}_{n-1}\|_2^2 + \mu_w[n]\big|v_w[n]\big|^2} < 1 \ . \tag{3.68}$$

Again, the two relations are coupled via the noise terms – the adaptation processes are not independent. As long as the step-sizes are smaller than the limits $\bar{\mu}_g[n]$ and $\bar{\mu}_w[n]$, local stability is guaranteed no matter how large the noise terms $v_g[n]$ and $v_w[n]$ are.

In each iteration step the error-energies, the ℓ_2-norm of the parameter error-vector and the squared undisturbed a-priori error are guaranteed to remain smaller than the disturbance energy, the ℓ_2-norm of the parameter error-vector at the previous iteration-step with the squared noise terms. The noise terms contain the undisturbed a-priori errors of the respectively other system, cf. (3.61) and (3.62). Therefore, no matter how large these errors are, the error-energies remain bounded if the conditions (3.65) and (3.66) are satisfied.

Feedback Structure – Local Passivity For the derivation of the feedback structure and the global stability analysis, the update equations in the error-vector form (3.54) and (3.55) are combined by defining the following vectors

$$\boldsymbol{\varphi}_n \triangleq [-\mathbf{x}_n', \boldsymbol{\psi}_n] \ , \tag{3.69}$$

$$\tilde{\mathbf{h}}_n \triangleq [\tilde{\mathbf{g}}_n'^T, \tilde{\mathbf{w}}_n^T]^T \ , \tag{3.70}$$

and the positive definite and symmetric step-size matrix $\mathbf{M}_n$. The update equation in error-vector form reads

$$\tilde{\mathbf{h}}_n = \tilde{\mathbf{h}}_{n-1} - \mathbf{M}_n\big(e_a[n] + v_e[n]\big)\boldsymbol{\varphi}_n^H \ , \tag{3.71}$$

whereby the undisturbed a-priori error is

$$e_a[n] = -e_{a,g} + e_{a,w}[n] = \boldsymbol{\varphi}_n\tilde{\mathbf{h}}_{n-1} \ . \tag{3.72}$$

The passivity relation is (derivation see Appendix A.3)

$$\frac{\tilde{\mathbf{h}}_n^H\mathbf{M}_n^{-1}\tilde{\mathbf{h}}_n + \big|e_a[n]\big|^2}{\tilde{\mathbf{h}}_{n-1}^H\mathbf{M}_n^{-1}\tilde{\mathbf{h}}_{n-1} + \big|v_e[n]\big|^2} < 1 \quad \text{if } 0 < \boldsymbol{\varphi}_n\mathbf{M}_n\boldsymbol{\varphi}_n^H < 1 \ . \tag{3.73}$$

It is assumed that the time-variant step-size matrix $\mathbf{M}_n$ can be decomposed in a (positive) time-variant scalar $\alpha[n]$ and a constant, symmetric and positive definite matrix $\mathbf{M}$, i.e.,

$$\mathbf{M}_n = \alpha[n]\mathbf{M} \ . \tag{3.74}$$

The update equation is

$$\mathbf{h}_n = \mathbf{h}_{n-1} + \alpha[n]\mathbf{M}\tilde{e}_a[n]\boldsymbol{\varphi}_n^H \ , \tag{3.75}$$

with $\tilde{e}_a[n] = \boldsymbol{\varphi}_n\mathbf{h}_{n-1} - x[n] = e_a[n] + v_e[n]$. With the special step-size matrix $\bar{\mathbf{M}}_n = \bar{\alpha}[n]\mathbf{M}$, whereby $\mathbf{M}$ is, e.g., a diagonal matrix with positive entries, and $\bar{\alpha}[n]$ satisfies the condition

$$\bar{\alpha}[n] = \frac{1}{\boldsymbol{\varphi}_n\mathbf{M}\boldsymbol{\varphi}_n^H} = \|\boldsymbol{\varphi}_n\|_{\mathbf{M}}^{-2} \ , \tag{3.76}$$

the update equation in error-vector form reads

$$\tilde{\mathbf{h}}_n = \tilde{\mathbf{h}}_{n-1} - \underbrace{\bar{\alpha}[n]\mathbf{M}}_{=\bar{\mathbf{M}}_n}\big(e_a[n] + \bar{v}_e[n]\big)\boldsymbol{\varphi}_n^H \ . \tag{3.77}$$

Here,

$$\bar{v}_e[n] = \frac{\alpha[n]}{\bar{\alpha}[n]}v_e[n] - \underbrace{\left(1 - \frac{\alpha[n]}{\bar{\alpha}[n]}\right)e_a[n]}_{\text{feedback path}} \tag{3.78}$$

is a new noise term, which defines the feedback path. The local passivity relation is

$$\frac{\tilde{\mathbf{h}}_n^H\mathbf{M}^{-1}\tilde{\mathbf{h}}_n + \bar{\alpha}[n]\big|e_a[n]\big|^2}{\tilde{\mathbf{h}}_{n-1}^H\mathbf{M}^{-1}\tilde{\mathbf{h}}_{n-1} + \bar{\alpha}[n]\big|\bar{v}_e[n]\big|^2} = 1 \tag{3.79}$$

since the step-size in (3.77) is $\bar{\mathbf{M}}_n$ which satisfies the condition $\boldsymbol{\varphi}_n\bar{\mathbf{M}}_n\boldsymbol{\varphi}_n^H = 1$.

The forward map $\bar{\mathbb{T}}_n$, which maps the input signals $\big\{\mathbf{M}^{-1/2}\tilde{\mathbf{h}}_{n-1}, \sqrt{\bar{\alpha}[n]}\bar{v}_e[n]\big\}$ to the output signals $\big\{\mathbf{M}^{-1/2}\tilde{\mathbf{h}}_n, \sqrt{\bar{\alpha}[n]}e_a[n]\big\}$, is lossless, $\big\|\bar{\mathbb{T}}_n\big\|_2 = 1$, as the passivity relation (3.79) shows. The feedback path is defined in (3.78). Since the gain of the forward path is one, the algorithm (3.71) with the step-size matrix $\mathbf{M}_n = \alpha[n]\mathbf{M}$ is locally stable if the gain of the feedback path is less than one. Hence, for local stability

$$\left|1 - \frac{\alpha[n]}{\bar{\alpha}[n]}\right| < 1 \ , \tag{3.80}$$

which yields the bounds

$$0 < \alpha_n < 2\bar{\alpha}[n] \ . \tag{3.81}$$

Global Passivity Relations The stability of the algorithm (3.71) with the step-size $\mathbf{M}_n = \alpha[n]\mathbf{M}$ considering the time horizon $n = 0, \dots, N$ is investigated now. Rewriting the local relation (3.79)

$$\tilde{\mathbf{h}}_n^H\mathbf{M}^{-1}\tilde{\mathbf{h}}_n + \bar{\alpha}[n]\big|e_a[n]\big|^2 = \tilde{\mathbf{h}}_{n-1}^H\mathbf{M}^{-1}\tilde{\mathbf{h}}_{n-1} + \bar{\alpha}[n]\big|\bar{v}_e[n]\big|^2 \tag{3.82}$$

and summation of both sides over the finite time horizon $n = 0, \dots, N$ gives

$$\tilde{\mathbf{h}}_N^H\mathbf{M}^{-1}\tilde{\mathbf{h}}_N + \sum_{n=0}^{N}\bar{\alpha}[n]\big|e_a[n]\big|^2 = \tilde{\mathbf{h}}_{-1}^H\mathbf{M}^{-1}\tilde{\mathbf{h}}_{-1} + \sum_{n=0}^{N}\bar{\alpha}[n]\big|\bar{v}_e[n]\big|^2 \ . \tag{3.83}$$

Since the quadratic form $\tilde{\mathbf{h}}_N^H\mathbf{M}^{-1}\tilde{\mathbf{h}}_N > 0$ ($\mathbf{M}$ is positive definite) it holds that

$$\sum_{n=0}^{N}\bar{\alpha}[n]\big|e_a[n]\big|^2 \le \tilde{\mathbf{h}}_{-1}^H\mathbf{M}^{-1}\tilde{\mathbf{h}}_{-1} + \sum_{n=0}^{N}\bar{\alpha}[n]\big|\bar{v}_e[n]\big|^2 \ , \tag{3.84}$$

which, using the triangle inequality, yields

$$\sqrt{\sum_{n=0}^{N} \bar{\alpha}[n]\big|e_a[n]\big|^2} \leq \sqrt{\tilde{\mathbf{h}}_{-1}^{H}\mathbf{M}^{-1}\tilde{\mathbf{h}}_{-1}} + \sqrt{\sum_{n=0}^{N} \bar{\alpha}[n]\big|\bar{v}_e[n]\big|^2} \,. \tag{3.85}$$

Insertion of the expression (3.78) for the modified noise term yields the energy-relation (see Appendix A.4)

$$\sqrt{\sum_{n=0}^{N} \bar{\alpha}[n]\big|e_a[n]\big|^2} \leq \frac{1}{1-\gamma_N}\left(\sqrt{\tilde{\mathbf{h}}_{-1}^{H}\mathbf{M}^{-1}\tilde{\mathbf{h}}_{-1}} + \delta_N\sqrt{\sum_{n=0}^{N} \bar{\alpha}[n]\big|v_e[n]\big|^2}\right) \,, \tag{3.86}$$

with

$$\delta_N = \max_{n=0,\ldots,N} \frac{\alpha[n]}{\bar{\alpha}[n]} \tag{3.87}$$

$$\gamma_N = \max_{n=0,\ldots,N} \left|1 - \frac{\alpha[n]}{\bar{\alpha}[n]}\right| \,. \tag{3.88}$$

Equation (3.86) relates the error energy (left hand side) to the disturbance energy (right hand side). If

$$\gamma_N < 1 \tag{3.89}$$

the algorithm is globally stable. Hence, the bounds

$$0 < \alpha[n] < 2\bar{\alpha}[n] \quad n = 0, \ldots, N \tag{3.90}$$

yield also a globally stable behaviour of the algorithm (3.75) with the step-size $\mathbf{M}_n = \alpha[n]\mathbf{M}$.

Example 1: Identification of the Linear Part of a Wiener System – Local Stability In this example the linear filter (FIR) of a Wiener-system with real valued input signal and real valued system parameters is identified with the proposed method. The algorithm in (3.51) and (3.52) is used for identification. Different step-sizes are chosen and their effect on the stability of the algorithm is determined.

The reference Wiener-system is a linear FIR-filter with 17 taps defining a bandpass filter and a static nonlinear function with a linear and a third order part, $f\big(z[n]\big) = z[n] - \frac{1}{9}z[n]^3$. The nonlinear filter $g\big(d[n]\big)$ is also a power series with only uneven terms up to the third order, $g\big(d[n]\big) = w_1 d[n] + w_2 d[n]^3$. The input signal $x[n]$ is drawn from a white, zero-mean gaussian process, $x[n] \sim \mathcal{N}(0, \sigma_x^2 = 1)$ with unit variance. It is assured that significant nonlinear distortion occurs, i.e., the magnitude of the linear filter output-signal $z[n]$ is large enough to fall in the saturation region of the nonlinear output-function. The algorithm is run 50 times and averages are taken to approximate the relative misadjustment

$$m_g[n] = \mathbb{E}\left(\frac{\|\tilde{\mathbf{g}}_n^{'}\|_2^2}{\|\tilde{\mathbf{g}}_{-1}^{'}\|_2^2}\right) \approx \frac{1}{50}\sum_{r=1}^{50} \frac{\|\tilde{\mathbf{g}}_n^{'}(r)\|_2^2}{\|\tilde{\mathbf{g}}_{-1}^{'}\|_2^2} \,. \tag{3.91}$$

Here, $\tilde{\mathbf{g}}_{-1}^{'}$ denotes the initial error-vector. The initial estimate is $\mathbf{g}_{-1}^{'} = \mathbf{0}$, $\tilde{\mathbf{g}}_n^{'}(r)$ denotes the error-vector at iteration-step n for the realisation r. The initial estimate for the nonlinear adaptive filter $g(\cdot)$ is $\mathbf{w}_{-1} = [1, 0]^T$. No measurement noise $v[n]$ is added, the remaining misadjustment of the linear filter parameter is therefore entirely due to the nonlinear saturation characteristic at the output of the Wiener system.

Case 1 In this case the step-sizes are $\mu_g[n] = 0,001$ and $\mu_w[n] = 0$, thus effectively no nonlinear filter $g(\cdot)$ is placed at the output of the Wiener system, the nonlinear distortions remain uncompensated. A relative misadjustment of approx. -20 dB is achieved, see Fig. 3.6. The system is stable. Since no measurement noise is added at the output of the Wiener-system, the misadjustment is only due to the nonlinear effects.

Case 2 The step-size $\mu_w = 0,01$ in this case, the step-size $\mu_g[n] = \bar{\mu}_g[n]$. The adaptive nonlinear filter $g(\cdot)$ compensates the effects of the non-linearity $f(\cdot)$ and a significantly better system identification of the linear filter is achieved. A relative misadjustment of approx. -30 dB is achieved, an improvement of approx. 10 dB, compared to case 1. The system is on the stability bound for guaranteed locally stable behaviour. The learning-curves for this case and case 1 are illustrated in Fig. 3.6. A significantly different learning behaviour, compared to case 1, can be observed. After a very fast initial phase, the learning becomes slower but continues, whereas in case 1 the learning process reaches the steady-state after approx. 5000 iterations.

Case 3 The time-variant step-size $\mu_g[n]$ is reduced to $\mu_g[n] = 0,5\bar{\mu}_g[n]$, the step-size $\mu_w[n] = 0,01$. The identification is not as accurate as in case 2, but the learning-process is smoother due to the smaller step-size $\mu_g[n]$ and the algorithm is guaranteed to be locally stable. The achieved misadjustment of approx. -26 dB is still significantly smaller than the misadjustment achieved in case 1. Fig. 3.7 shows the learning curve for this case, together with the learning-curve for case 2.

Case 4 The step-size $\mu_g[n] = 0,01$ again, but now $\mu_g[n] = 1,5\bar{\mu}_g[n]$. The algorithm is not guaranteed to be locally stable. Compared with case 2 a larger misadjustment results, together with a relatively turbulent learning behaviour, see Fig. 3.8.

Case 5 Further increase of the step-size $\mu_g[n]$ to the $\mu_g[n] = 2\bar{\mu}_g[n]$ results in an unstable system, see Fig. 3.8.

Case 6 A step-size $\mu_w[n] = 1,5$ together with the step-size $\mu_g[n] = \bar{\mu}_g[n]$ results also in an unstable system (without illustration).

Example 2: Identification of the Linear Part of a Wiener System – Global Stability
In this example a Wiener system with a complex-valued input signal $x[n]$, drawn from a zero mean, white gaussian process $x[n] \sim \mathcal{N}(0, \mathbf{R}_{xx} = \mathbf{I})$ is identified with the algorithm (3.75). The correlation matrix of the complex-valued input signal is given as $\mathbf{R}_{xx} = \mathbb{E}\{\mathbf{x}^H\mathbf{x}\}$ with $\mathbf{x}^T = \Big[\Re\{x[n]\}, \Im\{x[n]\}\Big]$. The linear filter of the reference Wiener system is a 17 taps linear FIR bandpass filter, the static nonlinear filter of the reference system is given as $f\big(z[n]\big) = z[n] - \frac{1}{20}\big(1+j0,01\big)z[n]\big|z[n]\big|^2$. Nonlinear AM/AM and AM/PM conversion occur in the static nonlinear filter. The nonlinear filter intended to compensate these nonlinear effects if also of order three, $g\big(d[n]\big) = w_1 d[n] + w_2 d[n]\big|d[n]\big|^2$. The constant step-size matrix is given as

$$\mathbf{M} = \begin{pmatrix} c_g\mathbf{I} & \mathbf{0} \\ \mathbf{0} & c_w\mathbf{I} \end{pmatrix}. \tag{3.92}$$

Different constants c_g and c_w define the adaptation behaviour of the linear part $\mathbf{g}'$ and the nonlinear part $\mathbf{w}$, respectively. The initial values for the adaptive algorithm are $\mathbf{h}_{-1}^T =$

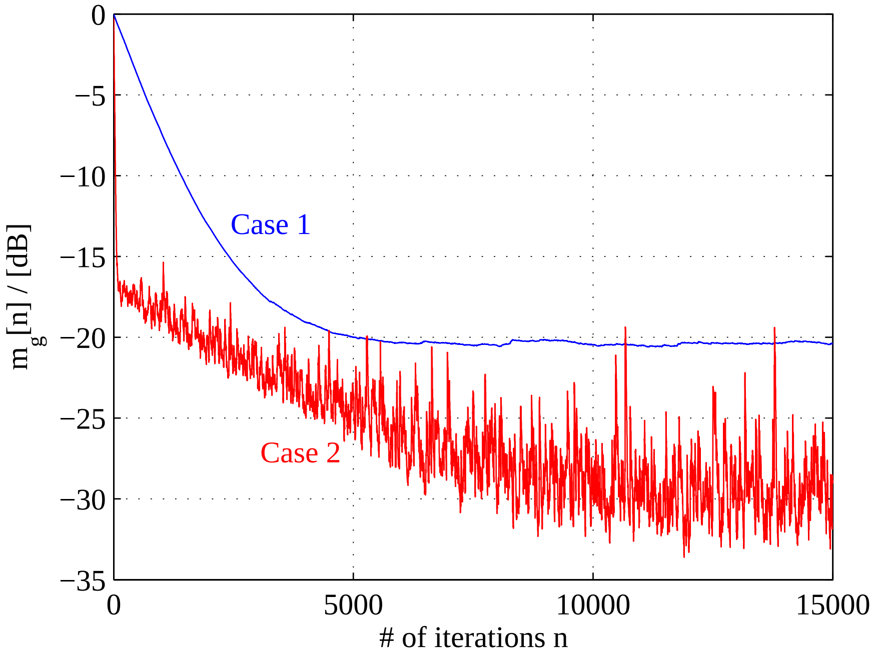


Figure 3.6: Relative misadjustment of the adaptive identification of the linear filter of a real-valued Wiener system. Case 1 is without the compensation of the nonlinear effects.

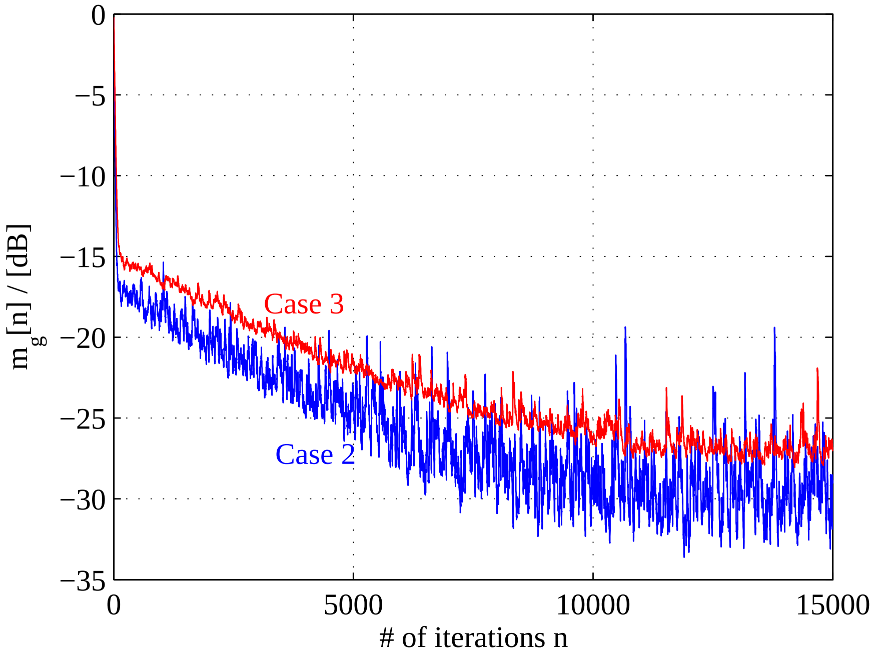


Figure 3.7: Relative misadjustment of the adaptive identification of the linear filter of a real-valued Wiener system. Case 3 is with a smaller step-size $\mu_g[n]$ compared to case 2.

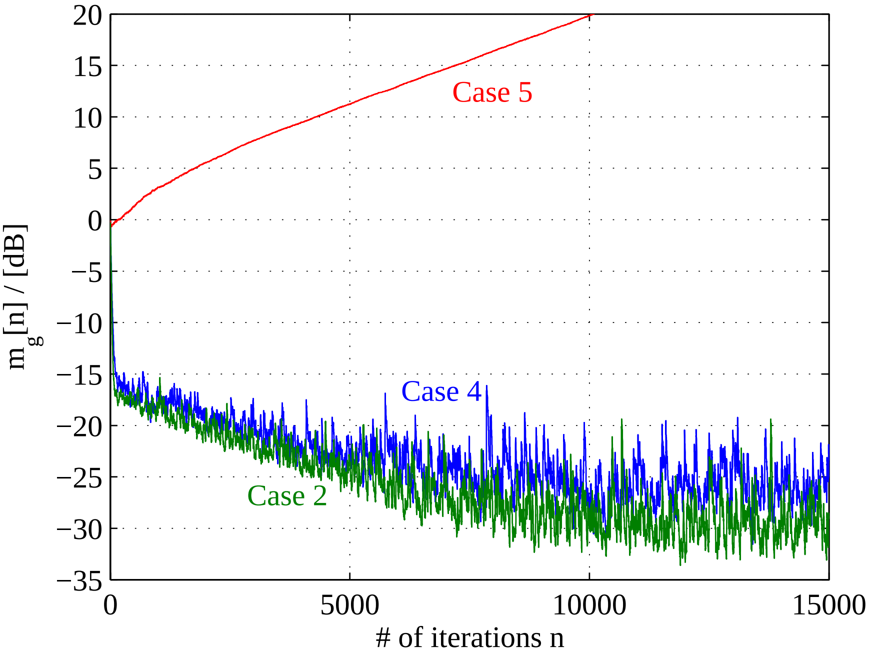


Figure 3.8: Relative misadjustment of the adaptive identification of the linear filter of a real-valued Wiener system. Case 4 is with a larger step-size compared to case 2, case 5 is with a step-size of $\mu_g[n] = 2\bar{\mu}_g[n]$.

$\big[\underbrace{0,\ldots,0}_{=\mathbf{g}'^T_{-1}}, \underbrace{1,0}_{=\mathbf{w}^T_{-1}}\big]$. Again, 50 simulation runs are performed and the average is taken. No noise $v[n]$ is added, thus the misadjustment in steady-state is only due to the nonlinear distortions of the filter $f(\cdot)$.

Case 1 In this case $c_g = 1$ and $c_w = 0$, thus the linear filter is identified without the compensation of the nonlinear effects. The factor $\alpha[n] = \bar{\alpha}[n]$. The algorithm is locally and globally stable. A misadjustment of approx. -24 dB is achieved, as Fig. 3.9 illustrates.

Case 2 Now $c_g = 1$ and $c_w = 0,08$ – the nonlinear effects are compensated with the nonlinear adaptive filter. Again, $\alpha[n] = \bar{\alpha}[n]$, resulting in a stable system. Significantly better identification is achieved as in case 1, see Fig. 3.9.

Case 3 If c_w is increased to $c_w = 0,2$ and $\alpha[n] = \bar{\alpha}[n]$, again a stable algorithm is obtained. Since the step-size is larger now, a higher misadjustment results in the steady-state, as can be seen in Fig. 3.9.

Case 4 Increasing $\alpha[n]$ further to $\alpha[n] = 1,99\,\bar{\alpha}[n]$ and $c_g = 1, c_w = 0,1$ results in a stable system as Fig. 3.10 shows. The stability bounds (3.90) are never exceeded. The steady-state misadjustment is significantly larger compared to the cases 2, and 3.

Case 5 If $\alpha[n] = 2\,\bar{\alpha}[n]$ the upper stability bound (3.90) is exceeded, resulting in an unstable algorithm, as Fig. 3.10 illustrates. The derived bounds (3.90) are tight.

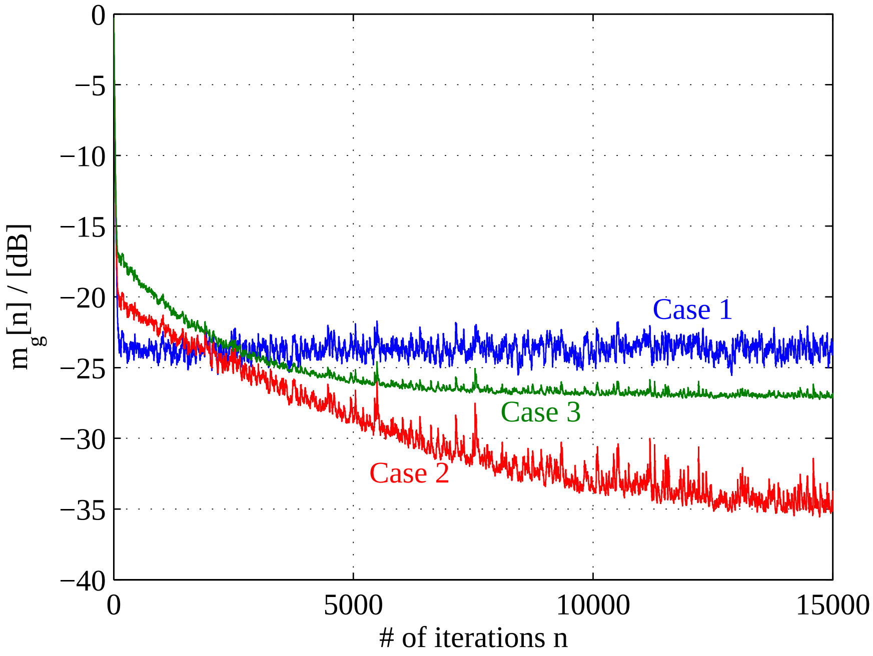


Figure 3.9: Relative misadjustment of the adaptive identification of the linear filter with complex-valued signal and system parameters. Case 1 is without the compensation of the nonlinear effects.

Second Step: Parameter Estimation for Static Nonlinear Filter

Once a first estimate for the linear filter parameters has been created using the described method, an identification algorithm for the nonlinear output-filter of the Wiener system can be started. The estimated linear filter remains constant during the estimation of the nonlinear part. Fig. 3.11 illustrates the situation. It is expected that the identification accuracy will suffer due to the non-perfect estimation of the linear filter. Therefore, after the first estimation of the nonlinear filter, a second step for improving the estimation of the linear filter parameters can follow.

The estimation of the parameters of the nonlinear filter is performed using a stochastic-gradient algorithm. If the nonlinear output-filter of the Wiener system is approximated with a series, the aim is to minimise the cost

$$J[n] = \mathbb{E}\Big\{\big|\tilde{e}[n]\big|^2\Big\} = \mathbb{E}\Big\{\big|d[n] - \hat{\boldsymbol{\phi}}_n \boldsymbol{\theta}\big|^2\Big\} \ , \tag{3.93}$$

where $\hat{\boldsymbol{\phi}}_n = \Big[\phi_1\big(\hat{z}[n]\big), \ldots, \phi_K\big(\hat{z}[n]\big)\Big]$, $\hat{z}[n]$ being the output of the estimated linear filter, $\hat{z}[n] = \hat{\mathbb{G}}\big(u[n]\big)$, $\{\phi_k(\cdot)\}_{k=0}^{K}$ is a set of basis-functions, e.g., powers, $\phi_k\big(z[n]\big) = z[n]|z[n]|^{2k}$, and $d[n]$ being the output of the Wiener system. Simplification of the expectation operator leads to the stochastic-gradient algorithm for the estimation of the parameters of the nonlinear filter:

$$\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + \mu[n]\tilde{e}_a[n]\hat{\boldsymbol{\phi}}_n^H \ , n \geq 0 \ , \boldsymbol{\theta}_{-1} \text{ given} \ . \tag{3.94}$$

Here, the disturbed a-priori error is

$$\tilde{e}_a[n] = d[n] - \hat{\boldsymbol{\phi}}_n \boldsymbol{\theta}_{n-1} = v[n] + f\big(z[n]\big) - \hat{\boldsymbol{\phi}}_n \boldsymbol{\theta}_{n-1} \ . \tag{3.95}$$

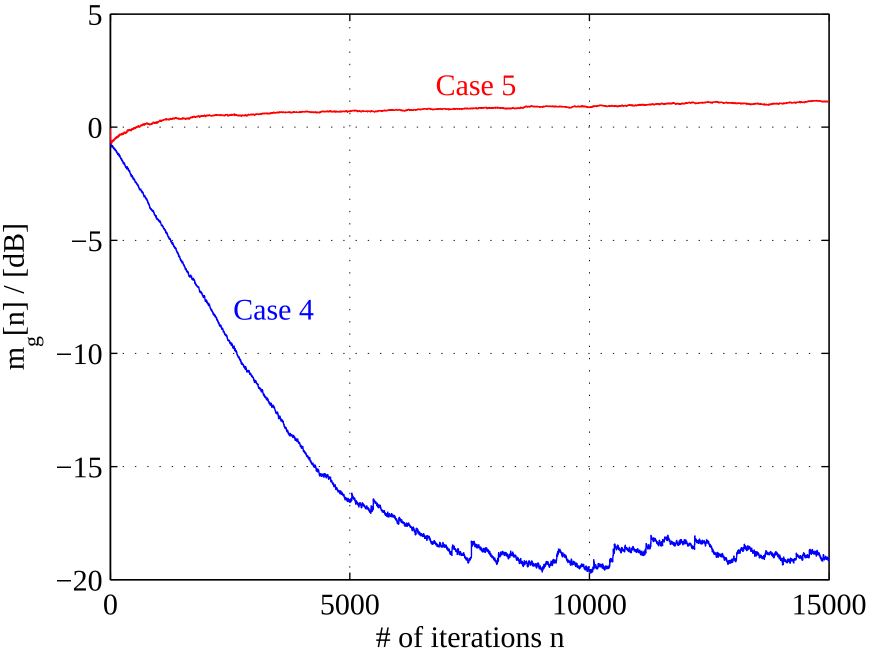


Figure 3.10: Relative misadjustment of the adaptive identification of the linear filter with complex-valued signal and system parameters. Case 5, where $\alpha[n] = 2\bar{\alpha}[n]$, leads to an unstable behaviour.

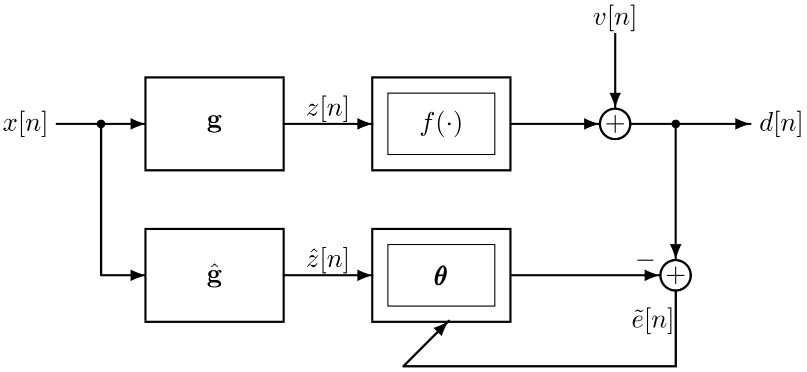


Figure 3.11: Adaptive identification of the nonlinear part of a Wiener system.

Assuming that the approximation with the specified set of basis-functions of the nonlinear output-filter of the Wiener system leads to an additional noise term

$$v_f[n] = f\big(z[n]\big) - \boldsymbol{\phi}_n \boldsymbol{\theta}_* \, , \tag{3.96}$$

whereby $\boldsymbol{\theta}_*$ denotes the optimal parameters in the mean-square sense and the row vector $\boldsymbol{\phi}_n$ is $\boldsymbol{\phi}_n = \Big[\phi_1\big(z[n]\big), \dots, \phi_K\big(z[n]\big)\Big]$, the disturbed a-priori error can be decomposed into

$$\tilde{e}_a[n] = v[n] + v_f[n] + \boldsymbol{\phi}_n \boldsymbol{\theta}_* - \hat{\boldsymbol{\phi}}_n \boldsymbol{\theta}_{n-1} \, . \tag{3.97}$$

Further,

$$\begin{aligned}\tilde{e}_a[n] &= v[n] + v_f[n] + \boldsymbol{\phi}_n\boldsymbol{\theta}_* - \hat{\boldsymbol{\phi}}_n\boldsymbol{\theta}_* + \hat{\boldsymbol{\phi}}_n\boldsymbol{\theta}_* - \hat{\boldsymbol{\phi}}_n\boldsymbol{\theta}_{n-1} \\ &= v[n] + v_f[n] + \big(\boldsymbol{\phi}_n - \hat{\boldsymbol{\phi}}_n\big)\boldsymbol{\theta}_* + \hat{\boldsymbol{\phi}}_n\big(\boldsymbol{\theta}_* - \boldsymbol{\theta}_{n-1}\big) \\ &= v[n] + v_f[n] + \tilde{\boldsymbol{\phi}}_n\boldsymbol{\theta}_* + \hat{\boldsymbol{\phi}}_n\tilde{\boldsymbol{\theta}}_{n-1} \\ &= v_\theta[n] + e_a[n]\end{aligned} \tag{3.98}$$

with

$$v_\theta[n] = v[n] + v_f[n] + v_g[n] \tag{3.99}$$

$$v_g[n] = \tilde{\boldsymbol{\phi}}_n\boldsymbol{\theta}_* \tag{3.100}$$

$$e_a[n] = \hat{\boldsymbol{\phi}}_n\tilde{\boldsymbol{\theta}}_{n-1}\ . \tag{3.101}$$

If it is assumed that the linear filter $\mathbb{G}(q^{-1})$ is identified perfectly, thus $\tilde{\boldsymbol{\phi}}_n = \mathbf{0}$, and, consequently, $v_g[n] = 0$. Only the measurement-noise $v[n]$ remains, as well as the disturbance $v_f[n]$, which is due to the approximation of the nonlinear filter with a truncated series. In general, the linear filter is not perfectly identified, thus the disturbance $v_g[n]$ does not vanish.

With these definitions, the deterministic robustness analysis of the gradient-algorithm (in error-vector form)

$$\tilde{\boldsymbol{\theta}}_n = \tilde{\boldsymbol{\theta}}_{n-1} - \mu[n]\big(e_a[n] + v_\theta[n]\big)\hat{\boldsymbol{\phi}}_n^H\ , \tag{3.102}$$

can be performed in a similar way as for linear-in-parameter models, see Section 3.1.1. This analysis leads to the bounds for the step-size

$$0 < \mu[n] < 2\bar{\mu}[n]\ ,\ n = 0, \ldots, N\ ,\ \bar{\mu}[n] = \frac{1}{\|\hat{\boldsymbol{\phi}}_n\|_2^2} \tag{3.103}$$

which assure global stability.

Example 1 continued: Identification of the Nonlinear Part of a Wiener System A relative misadjustment of $-20\,\text{dB} - -30\,\text{dB}$ was achieved in the estimation of the parameters of the linear filter of the first example, see Fig. 3.6. Having this estimate, the nonlinear part of the system is identified with a standard gradient algorithm, cf. (3.94). Again 50 simulation runs for different values of the misadjustment for the linear filter are performed. For this identification an a-posteriori LMS algorithm with $\alpha = 1$ is used, see Tab. 3.1, since this type of algorithm achieved the best results in Section 3.1.1, see Fig. 3.2. The a-posteriori LMS algorithm has further the advantage to be guaranteed stable.

Fig. 3.12 shows that the algorithm performs well, resulting in an accurate estimation of the nonlinear filter parameters if the identification of the linear part is accurate. Although the linear system is not estimated perfectly with a relative misadjustment of, e.g., $m_g = -30\,\text{dB}$, a relative misadjustment for the nonlinear filter parameters of $-50\,\text{dB}$ can be achieved. If the estimation of the linear filter parameters is less accurate, e.g., if a misadjustment of only $-20\,\text{dB}$ is achieved (no compensation of the output non-linearity, cf. Fig. 3.6, the achievable misadjustment is approx. 10 dB higher. It can be seen that the error in the identification of the linear filter parameters is propagated into the identification of the nonlinear filter.

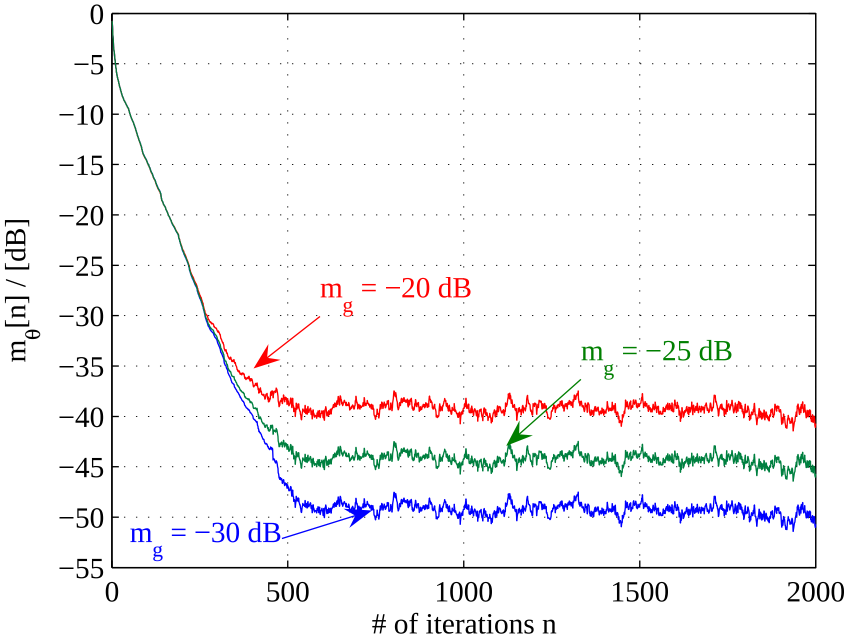


Figure 3.12: Relative misadjustment for the adaptive identification of the nonlinear part of the example Wiener system.

The Complete Algorithm

The complete algorithm is briefly summarised in the Tab. 3.3:

1. In the first step, the parameters of the linear filter are estimated. The update equations are given in (3.51) and (3.52). Local stability can be guaranteed if the step-sizes obey the conditions presented in (3.65) and (3.66). Global stability of the algorithm can be guaranteed for the special case of the matrix step-size $\alpha[n]\mathbf{M}$ (update equation (3.75)), if $\alpha[n]$ obeys the condition in (3.90).

2. In the second identification step, the nonlinear part is identified with the gradient algorithm (3.94). The bounds for the step-size for stability are presented in (3.103).

If the quality of the identification is not high enough after completing the two identification steps, the procedure can be repeated.

3.2 Discussion

In this chapter adaptive algorithms for the identification of dynamic nonlinear systems were discussed. Specifically, gradient-type algorithms for the parameter identification of linear-in-parameter models, such as Volterra models, and a model which is nonlinear-in-parameters, namely the Wiener model, were investigated with respect to robustness in the ℓ_2-sense. The analysis of the adaptive algorithms for linear-in-parameter models is straightforward and follows the same lines as the analysis for adaptive linear filters [71, 13].

The identification of a nonlinear Wiener system with unknown output non-linearity is performed in two steps: first, the linear filter is identified whereby the influence of the nonlinear

Step	identified part	up-date equation(s)	ad notation
1	linear filter	$\mathbf{g}'_n = \mathbf{g}'_{n-1} - \mu_g[n]\tilde{e}_a[n]\mathbf{x}'^H_n$ $\mathbf{w}_n = \mathbf{w}_{n-1} + \mu_w[n]\tilde{e}_a[n]\boldsymbol{\psi}^H_n$ $\tilde{e}_a[n] = x[n] + \mathbf{x}'_n\mathbf{g}'_{n-1} - \boldsymbol{\psi}_n\mathbf{w}_{n-1}$ *locally stable if:* $0 < \mu_g[n] < \|\mathbf{x}_n\|_2^{-2}$ $0 < \mu_w[n] < \|\boldsymbol{\psi}_n\|_2^{-2}$	$\mathbf{g}'$ . . . linear filter parameter $\mathbf{w}$ . . . inverse output filter $\mathbf{x}'_n$ . . . input of filter $\mathbf{g}'$ $\boldsymbol{\psi}_n$ . . . input of filter $\mathbf{w}$ cf. Fig. 3.5
	matrix step-size	$\mathbf{h}_n = \mathbf{h}_{n-1} + \alpha[n]\mathbf{M}\tilde{e}_a[n]\boldsymbol{\varphi}^H_n$ $\tilde{e}_a[n] = \boldsymbol{\varphi}_n\mathbf{h}_{n-1} - x[n]$ *locally and globally stable if:* $0 < \alpha[n] < 2\|\boldsymbol{\varphi}_n\|_{\mathbf{M}}^{-2}\ ,\forall n$	$\mathbf{h} = [\mathbf{g}'^T, \mathbf{w}^T]^T$ $\boldsymbol{\varphi}_n = [-\mathbf{x}'_n, \boldsymbol{\psi}_n]$ $\alpha[n]\mathbf{M}$ . . . matrix step-size
2	nonlinear filter	$\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + \mu[n]\tilde{e}_a[n]\hat{\boldsymbol{\phi}}^H_n$ $\tilde{e}_a[n] = d[n] - \hat{\boldsymbol{\phi}}_n\boldsymbol{\theta}_{n-1}$ *locally and globally stable if:* $0 < \mu[n] < 2\|\hat{\boldsymbol{\phi}}_n\|_2^{-2}\ ,\forall n$	$\boldsymbol{\theta}$ . . . nonlinear filter param. $\hat{\boldsymbol{\phi}}_n$ . . . input of filter $\boldsymbol{\theta}$ cf. Fig. 3.11

Table 3.3: The complete algorithm for the gradient identification of a Wiener system

part of the Wiener filter is compensated with a second adaptive nonlinear filter. Then, having this estimate, the nonlinear part of the Wiener system is identified. A deterministic analysis for the identification of the linear part was performed [16], providing tight bounds for the step-sizes guaranteeing locally and globally stable behaviour.

Chapter 4

Linearisation by Pre-distortion

After modelling the nonlinear system using one of the models presented in Chapter 2 and identification of the model-parameters, either using the direct least-squares method as in Chapter 2, or using an adaptive identification algorithm, discussed in Chapter 3, the next crucial task is to design a pre-filter (pre-distortion unit) which linearises the nonlinear system. Graphically, the problem can be represented as in Fig. 4.1. The nonlinear system $\mathbb{N}$ (the power amplifier with additional circuitry, e.g., modulators and mixers) is to be linearised with the nonlinear pre-filter $\mathbb{P}$. The targeted behaviour is a linear amplification

$$y[n] \approx d[n] = \mathbb{L}\big(u[n]\big) \ . \tag{4.1}$$

If the requirement is only to achieve a linear operation, the dispersion must not be compensated, i.e., $\mathbb{L} = \sum_{p=0}^{P-1} w_p q^{-1}$ is simply a linear filter – if the frequency dispersion is also compensated, the linear operator simplifies to a simple multiplication with a constant, in general a complex valued factor g, $\mathbb{L}\big(u[n]\big) = g \cdot u[n-\Delta]$, Δ being some delay. This is now also an equalisation problem. In the following sections the term *linearisation* refers in general to linearisation and equalisation. If $\mathbb{L} = \mathbb{I}$, the identity operator, the nonlinear system $\mathbb{N}$ is inverted and the pre-filter must be the inverse of $\mathbb{N}$, $\mathbb{P} = \mathbb{N}^{-1}$. In this case the gain $g = 1$ and $\Delta = 0$.

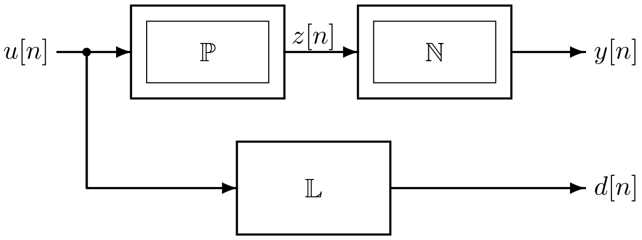


Figure 4.1: Linearisation problem

The problem is the design of the pre-filter $\mathbb{P}$, given the nonlinear filter $\mathbb{N}$, which is a more or less accurate model of a real physical system (the power amplifier with analogue circuitry for modulation and up-conversion), and the given targeted linear system $\mathbb{L}$. Since $\mathbb{N}$ is in general only a model of the real system, the linearisation will degrade if the model is not accurate. In Section 4.4.2 this problem is analysed by means of simulations.

In general, since the filter $\mathbb{N}$ is nonlinear, an analytical solution for the linearisation problem is not possible. Local solutions, i.e., a pre-filter $\mathbb{P}$ that linearises or inverts the nonlinear system $\mathbb{N}$ for a restricted class of input signals, can often be found. Formally, the nonlinear system $\mathbb{N}$ is linearised around a point $u_0[n]$ – the linearisation is accurate in a certain neighbourhood of this point, i.e.,

$$\left\|\mathbb{N}\Big(\mathbb{P}\big(u[n]\big)\Big) - \mathbb{L}\big(u[n]\big)\right\| < \delta \quad \text{if } u[n] \in \Big\{u[n]\Big|\big\|u_0[n] - u[n]\big\| < \varepsilon\Big\} . \tag{4.2}$$

E.g., for static nonlinearities and real-valued signals $u[n]$, the linearisation is accurate in a certain interval on the real line. In this case, the pre-filter can be implemented as a look-up table, see e.g., [72, 73]. The table entries are pre-calculated based on the model of the nonlinear system. The accuracy of the method is heavily dependent on the resolution of the look-up table – if the input signal is quantised using Q levels, the look-up table has Q entries. For complex signals Q^2 values have to be stored, which can be very large if the resolution is high, e.g., if $Q = 2^{12} = 4096$, corresponding to a 12 bit analogue-to-digital converter, the size of the look-up table is $16,8 \cdot 10^6$, requiring approx. 34 MB of storage if 2 bytes per sample are used. If memory effects have to be considered the size of the look-up table becomes very large – for the linearisation of dynamic nonlinear systems, look-up tables are prohibitively complex.

The inversion of general dynamic nonlinearities, expressed by a Volterra series, was pioneered by M. Schetzen in [4, 36], where a pth-order inverse of a Volterra system is derived. The system response of the tandem connection of the pth order inverse and the Volterra system is

$$\mathbb{N}\Big(\mathbb{P}\big(u[n]\big)\Big) = u[n] + \mathbb{Q}\big(u[n]\big) , \tag{4.3}$$

where $\mathbb{Q}\big(u[n]\big) = \sum_{i=p+1}^{\infty} \mathbb{Q}_i\big(u[n]\big)$ is a Volterra filter with the second through the pth-order homogeneous parts $\mathbb{Q}_i\big(u[n]\big)$ (cf., e.g., (2.26)) equal to zero. Thus, the nonlinear effects up to the pth order are compensated, but not the higher order nonlinear terms. The drawback of this approach is the relatively high complexity of the pre-inverse and the appearance of the high-order nonlinear terms beginning with the $p + 1$th term.

In this chapter, two iterative methods for the linearisation of a general Volterra filter are investigated – the first method is based on a fixed-point approach, presented in [18], the second approach is based on an alternative formulation of the linearisation problem [19, 74, 12]. Here, the idea is to modify the formulation of the linearisation problem to a root-search problem and to solve this problem iteratively. The advantage of this formulation is that fast iterative algorithms can be used, which reduce the required complexity compared with the fixed-point approach. The drawback of the two methods is the complex analysis of convergence. For the fixed-point approach a sufficient condition for convergence of the iterative procedure can be devised using the contraction-mapping theorem. Since this provides only sufficient conditions for convergence, they are in general too conservative – even if the conditions are not met, the algorithm may converge. Further, the verification of the convergence criterion is rather complex. In the investigated cases, based on models derived from measurements on power amplifiers, see Chapter 2, both methods showed good convergence behaviour with obvious restrictions on the targeted linear behaviour $\mathbb{L}$. A further alleviation is that the power amplifier nonlinearities are usually mild and the linearisation task is “easy” to achieve for the iterative techniques.

The iterative techniques are investigated with respect to convergence and speed using simulations. Here, realistic power amplifier models, either obtained from measurements or with the commercial simulation tool ADS, see Chapter 2, are used. In all cases (obvious) settings can be found that lead to stable behaviour.

Finally, measurement results show that the iterative techniques, applied to linearise a real microwave power amplifier, yield significant performance gains if compared to a simple input back-off, i.e., a reduction of the input power of the amplifier.

4.1 Formulation of the Linearisation Problem as an Optimisation Problem

The linearisation problem of Fig. 4.1 can be formulated in different ways. The objective is to achieve that the system output $y[n]$ using the pre-filter $\mathbb{P}$ is close to the desired signal $d[n] = \mathbb{L}(u[n]) = g \cdot u[n-\Delta]$, in the pre-distortion setting usually an amplification of the input signal $u[n]$. This can be formulated as an optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \left\| \mathbb{N}\Big(\mathbb{P}\big(\boldsymbol{\theta}, u[n]\big)\Big) - \mathbb{L}\big(u[n]\big) \right\| \ , \tag{4.4}$$

where an optimisation algorithm tries to find the optimal parameters $\boldsymbol{\theta}$ for the pre-filter. This formulation assumes that the structure of the pre-filter $\mathbb{P}$ is known – with nonlinear systems $\mathbb{N}$ this is in general not possible. The unknown pre-filter could be assumed, e.g., to be a Volterra filter of a certain structure (specific nonlinear order, specific kernel lengths). Fig. 4.2 shows a possible configuration using an adaptive algorithm for identification of the parameters of the pre-filter. The advantage here is that the nonlinear system $\mathbb{N}$ is not needed to be known. On the other hand, the specific pre-filter $\mathbb{P}$ can be a poor match to the unknown, exact pre-filter. Hence, even with the optimal parameters $\boldsymbol{\theta}$ of the pre-filter, only a poor linearisation quality can be achieved.

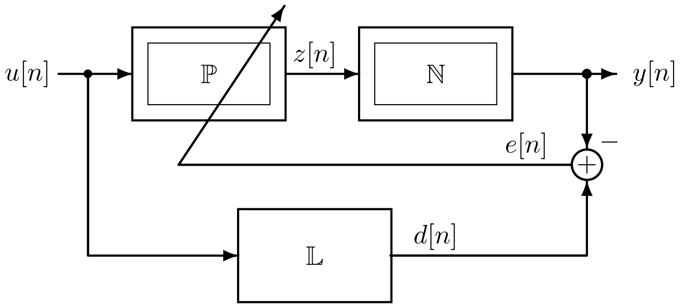


Figure 4.2: Linearisation problem – adaptive identification of the pre-filter without knowledge of the nonlinear system

The configuration in Fig. 4.3 is also often used. See e.g., [17], where this configuration is applied to linearise a Volterra system. Here, as in the previous configuration, see Fig. 4.2, a specific filter-structure $\mathbb{P}$ has to be assumed. The error $e[n]$ is

$$e[n] = \mathbb{P}\big(u[n]\big) - \mathbb{P}\left(\frac{1}{g} y[n]\right) \ . \tag{4.5}$$

If the error $e[n]$ vanishes,

$$\mathbb{P}\big(u[n]\big) = \mathbb{P}\left(\frac{1}{g} y[n]\right) \tag{4.6}$$

must hold for all signals $u[n]$. Hence,

$$y[n] = g \cdot u[n] \; . \tag{4.7}$$

In the general case the error signal will not vanish completely, giving

$$y[n] = g \cdot \mathbb{P}^{-1}\Big(\mathbb{P}\big(u[n]\big) - e[n]\Big) \; . \tag{4.8}$$

Consequently, even with optimal parameters for the pre-filter $\mathbb{P}$, the output is not a simple amplification of the input signal. The problem here is that the assumed filter $\mathbb{P}$ can be far from the optimal filter-structure for the pre-equaliser. Thus, even with optimal parameters of the filter $\mathbb{P}$, it is possible that only a poor linearisation quality can be achieved.

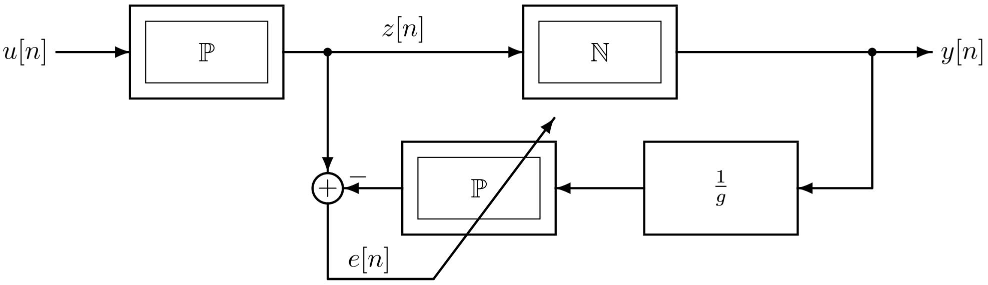


Figure 4.3: Linearisation problem – adaptive identification of the pre-filter without knowledge of the nonlinear system

4.2 The Fixed-Point Approach

Alternatively to the two presented methods, the linearisation problem can be formulated as a fixed-point equation – not with respect to the parameters $\boldsymbol{\theta}$ of a certain pre-filter $\mathbb{P}$, but a fixed-point equation in the signal $z[n]$ after the pre-filter [18]. This is conceptually different from the above presented optimisation approach. Now, the structure of the pre-filter must not be known beforehand, but it is assumed that the nonlinear filter $\mathbb{N}$ is given – in most cases as a model for the specific physical nonlinear system.

It is assumed that the nonlinear system $\mathbb{N}$ is a Volterra series – an assumption which is not very restrictive, since a large class of nonlinear systems can be modelled with Volterra series. If the signal after the pre-filter is $z[n] = \mathbb{P}\big(u[n]\big)$, the fixed-point equation can easily be derived:

$$y[n] = \mathbb{L}\big(u[n]\big) \Leftrightarrow \mathbb{N}\big(z[n]\big) = \sum_{p=1}^{P} \mathbb{N}_p\big(z[n]\big) = \mathbb{L}\big(u[n]\big) \; . \tag{4.9}$$

With the decomposition

$$\mathbb{N}_1\big(z[n]\big) + \sum_{p=2}^{P} \mathbb{N}_p\big(z[n]\big) = \mathbb{L}\big(u[n]\big) \; , \tag{4.10}$$

the signal after the pre-filter is

$$z[n] = \mathbb{N}_1^{-1}\left(\mathbb{L}\big(u[n]\big) - \sum_{p=2}^{P} \mathbb{N}_p\big(z[n]\big)\right) = \mathbb{T}_u\big(z[n]\big) \; . \tag{4.11}$$

Here, $\mathbb{N}_p$ are the homogeneous Volterra operators of order p and it is assumed that the first-order operator $\mathbb{N}_1$ (a linear filter) is invertible. The transformation $\mathbb{T}$ is indexed with the input signal $u[n]$ to emphasise the dependence on it (it depends also on $\mathbb{L}$). Here, in contrast to the above mentioned two methods, the nonlinear Volterra system $\mathbb{N}$, which is in practice the model for the nonlinear power amplifier, needs to be known. The essential difference in this approach is, that the signal $z[n]$ after the pre-filter $\mathbb{P}$ is identified, not the parameters $\boldsymbol{\theta}$ of a specific pre-filter. In (4.11) the signal $z[n]$ is a fixed-point of the transformation $\mathbb{T}_u$. Searching the fixed-point is thus equivalent to determining the pre-filter $\mathbb{P}$ itself. Since an analytic solution to the above fixed-point equation is rarely available, the fixed-point $z[n]$ is searched iteratively with successive approximation:

$$z_{i+1}[n] = \mathbb{T}_u\big(z_i[n]\big), \quad z_0[n] \text{ given} . \tag{4.12}$$

Whether the method converges to a solution $z[n]$ or not depends on the transformation $\mathbb{T}$, which depends on the input signal $u[n]$ and the targeted linear operation $\mathbb{L}$. Here a simple linear amplification with a delay Δ, $\mathbb{L}\big(u[n]\big) = g \cdot u[n-\Delta]$ is used. Applying the Contraction Mapping Theorem [49, 48], which states basically that the method of successive approximation converges if $\mathbb{T}$ is contractive (see Appendix B), it can be determined whether the method of successive approximation (4.12) is guaranteed to converge. If the transformation $\mathbb{T}$ is further differentiable and $\big\|\mathbb{T}'(z)\big\| \leq \alpha < 1$ on a convex set[1] $\mathscr{K}$, the transformation is a contraction mapping, since $\big\|\mathbb{T}(z_1) - \mathbb{T}(z_2)\big\| \leq \sup\big\|\mathbb{T}'(z)\big\|\|z_1 - z_2\| \leq \alpha\|z_1 - z_2\|$, using the mean-value inequality [48]. Therefore, it is guaranteed that the method converges.

For applying the Contraction Mapping Theorem for the linearisation problem, the problem must be formulated in a Banach space and, therefore, a specific norm has to be chosen. Many choices of norms are possible, e.g., any of the ℓ_p-sequence norms may be chosen. Once a norm has been chosen, it must be confirmed that the transformation $\mathbb{T}_u$ is contractive. In order to confirm this, the norms of the homogeneous Volterra operators $\mathbb{N}_p$, induced by the chosen signal norm, have to be computed. In practice, these norms cannot be computed – computable upper bounds have to be used, which may be tight or loose. Hence, using different signal norms, different restrictions, necessary to guarantee the contractiveness, are produced [18]. Since the Contraction Mapping Theorem gives sufficient conditions for convergence, the obtained convergence results may be very loose and seldom possible to fulfill in practice. The complicated nature for proving convergence and the loose bounds obtained are not very helpful in applications – therefore, the convergence is investigated in Section 4.4 using simulations.

Regarding the convergence rate of the successive approximation procedure applied to a contraction mapping $\mathbb{T}$ with the fixed-point $z[n]$ the inequality

$$\Big\|\mathbb{T}\big(z_{i-1}[n]\big) - \mathbb{T}\big(z[n]\big)\Big\| = \big\|z_i[n] - z[n]\big\| \leq \alpha\big\|z_{i-1}[n] - z[n]\big\| \tag{4.13}$$

results. The sequence $\{z_i[n]\}$ converges thus only linearly to the fixed-point $z[n]$ since

$$\lim_{j\to\infty} \sup_{i\geq j} \frac{\big\|z_i[n] - z[n]\big\|}{\big\|z_{i-1}[n] - z[n]\big\|} = \alpha \tag{4.14}$$

for some $0 < \alpha < 1$. The slow convergence is the main drawback of this linearisation method.

Slow convergence means that for good accuracy of linearisation a large number of iterations has to be performed. A large number of iterations means a high complexity – either a pipelined

---

[1] A set $\mathscr{K}$ in a linear vector space is said to be convex if, given $x_1, x_2 \in \mathscr{K}$, all points of the form $\beta x_1 + (1-\beta)x_2$ with $0 \leq \beta < 1$ are in $\mathscr{K}$.

approach with a large number of stages or a high operating frequency has to be used. E.g., if five iterations are necessary, five equal stages have to be implemented. Alternatively, one stage has to be executed five times during one sampling interval. Fig. 4.4 shows a graphical representation of the successive approximation method with three stages. The filter $\mathbb{G}$ is, as can be seen form (4.11),

$$\mathbb{G}(\cdot) = \mathbb{N}_1^{-1}\left(\sum_{p=2}^{P}\mathbb{N}_p(\cdot)\right) \tag{4.15}$$

assuming that $\mathbb{N}_1$ is linear and invertible.

For each iteration the filter $\mathbb{G}$ has to be evaluated – depending on the complexity of the Volterra model $\mathbb{N}$ this requires possibly a lot of hardware resources. The complexity can be lowered either by using low complex models, requiring only few calculations, e.g., only static models or Wiener-or Hammerstein models, or by increasing the convergence rate using other, faster methods. This later approach has the advantage that still accurate models with a relative high complexity can be used.

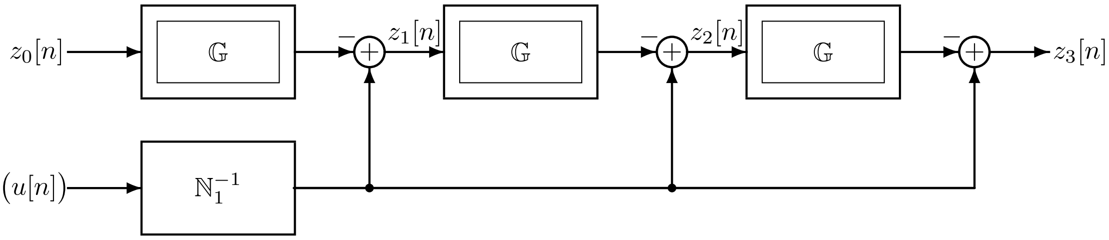


Figure 4.4: Linearisation with successive approximation – three iterations

4.3 Linearisation using the Secant Method

The successive approximation method, discussed in Section 4.2 suffers from a slow convergence rate – the convergence rate is only linear. Depending on the specific non-linearity it may be necessary to use a high number of iterations in order to achieve a small linearisation error. This increases the requirements on the hardware – either a fast device, e.g., a DSP or an FPGA clocked by a high frequency, or a large device, e.g., an FPGA with a high number of gates, have to be used. Increasing the convergence rate is therefore essential for a real-time implementation.

The origin of the slow convergence is the fixed-point formulation of the linearisation problem which implies the usage of the successive approximation method with only linear convergence rate. A natural and straightforward formulation of the linearisation problem is of the form

$$\mathbb{N}\big(z[n]\big) - \mathbb{L}\big(u[n]\big) = \mathbb{T}_u\big(z[n]\big) = 0\ , \tag{4.16}$$

derived from Fig. 4.1 or from (4.9). This equation has to be solved for the signal $z[n]$. Again, as in the case of the fixed-point equation (4.11), an analytic solution is most often not achievable. Iterative techniques have to be used, and now, in contrast to the fixed-point formulation, a large number of techniques are available, see, e.g., [75]. Among them, Newton's method is the most popular and the most powerful method. As originally conceived, it applies to equations of a single real variable but it can be extended to nonlinear transformations on normed spaces [48]. In order to solve the above equation (4.16) with Newton's method, the nonlinear transformation $\mathbb{T}_u$ must be differentiable. If this is assured, Newton's method reads

$$z_{i+1}[n] = z_i[n] - \big[\partial_z \mathbb{T}_u\big(z_i[n]\big)\big]^{-1} \mathbb{T}_u\big(z_i[n]\big) \ , \tag{4.17}$$

using some starting value $z_0[n]$. Newton's method converges quadratically, meaning that

$$\big\|z_{i+1}[n] - z[n]\big\| \leq c\big\|z_i[n] - z[n]\big\|^2 \ , \tag{4.18}$$

$z[n]$ denoting a solution of (4.16).

The Newton step

$$\delta z_i[n] = -\big[\partial_z \mathbb{T}_u\big(z_i[n]\big)\big]^{-1} \mathbb{T}_u\big(z_i[n]\big) \ , \tag{4.19}$$

where $\delta z_i[n] = z_{i+1}[n] - z_i[n]$, decreases also the cost function

$$J\big(z[n]\big) = \Big\|\mathbb{T}_u\big(z[n]\big)\Big\|_2^2 = \mathbb{T}_u^H\big(z[n]\big)\mathbb{T}_u\big(z[n]\big) \ . \tag{4.20}$$

This can be seen if the Newton step is multiplied with the gradient of the cost function $\partial_z J\big(z[n]\big)$, which is

$$\partial_z J\big(z[n]\big) = \partial_z \Big[\mathbb{T}_u^H\big(z[n]\big)\mathbb{T}_u\big(z[n]\big)\Big] = \mathbb{T}_u^H\big(z[n]\big)\partial_z\mathbb{T}_u\big(z[n]\big). \tag{4.21}$$

Multiplication of the gradient at the point $z_i[n]$ with the Newton step $\delta z_i[n]$ yields

$$\begin{aligned}\partial_z J\big(z_i[n]\big)\delta z_i[n] &= \Big(\mathbb{T}_u^H\big(z[n]\big)\partial_z\mathbb{T}_u\big(z_i[n]\big)\Big)\Big(-\Big[\partial_z\mathbb{T}_u\big(z_i[n]\big)\Big]^{-1}\mathbb{T}_u\big(z_i[n]\big)\Big)\\ &= -\mathbb{T}_u^H\big(z_i[n]\big)\mathbb{T}_u\big(z_i[n]\big) < 0 \ ,\end{aligned} \tag{4.22}$$

proving that the cost (4.20) is indeed reduced with every step. This means that every solution of (4.16) minimises also the cost function (4.20). But the converse is not necessarily true, since there might be local minima of (4.20) which are not solutions of (4.16). This is the fundamental and important difference between the optimisation approach where an objective function is minimised and the direct approach presented above, where a solution for a nonlinear (system of) equation(s) is searched for.

The fast convergence rate of Newton's method has a price attached: its large complexity [74]. Now, for one iteration, not only the transformation $\mathbb{T}_u$ has to be evaluated at $z_i[n]$. Additionally, the Jacobian $\partial_z\mathbb{T}_u\big(z_i[n]\big)$ has to be evaluated and inverted. The principal problem with this method is the computation of the Jacobian, which might not be easy or even possible – the transformation $\mathbb{T}$ has to be differentiable, which, for the specific nonlinear models discussed in Chapter 2 is not the case, since terms involving the magnitude of the signal $\big|z[n]\big|$ are appearing – these terms are only analytic for $z[n] = 0$. See, e.g., the general equivalent baseband Volterra series in (2.26). A solution could be to exclude these terms in the modelling, but this results in a poor model-system match. It is expected that the output is dependent on the actual magnitude of the input signal, the restriction being thus not well-founded.

The main difficulty in applying the fast Newton's method for solving (4.16) is thus the determination and evaluation of the Jacobian. Every simplification will further result in a slower procedure. The aim is thus to simplify Newton's method but keep the convergence rate as high as possible. A practical and for the purpose well-suited method is the secant method. The secant method for the specific problem (4.16) is

$$z_{i+1}[n] = z_i[n] - \frac{z_i[n] - z_{i-1}[n]}{\mathbb{N}\big(z_i[n]\big) - \mathbb{N}\big(z_{i-1}[n]\big)} \mathbb{T}_u\big(z_i[n]\big) \ , \tag{4.23}$$

with two (different) starting values $z_{-1}[n]$ and $z_0[n]$. A convenient choice is $z_{-1}[n] = 0$ and $z_0[n] = u[n]$, since it is expected that the signal after the pre-filter differs only slightly from the original input signal $u[n]$.

The computation of the Jacobian is thus no longer required. The only condition on the transformation $\mathbb{N}$ is to be continuous, a requirement which is naturally fulfilled for the models discussed in Chapter 2. The above equation (4.23) calculates the output of the pre-filter $\mathbb{P}$ for each time instant n. If the nonlinear transformations $\mathbb{N}$ and $\mathbb{T}_u$ have memory, they are time-variant since the outputs $\mathbb{N}\big(z_i[n]\big)$, $\mathbb{N}\big(z_{i-1}[n]\big)$, and $\mathbb{T}_u\big(z_i[n]\big)$ depend on the previously calculated pre-filter outputs $z[n-1], \ldots, z[n-M+1]$ (assuming a memory length of M samples), as well as the pre-filter input signal $u[n]$.

For a formal derivation of the secant method, the general case with signals of length M, e.g., $z[n] = \big\{z[n], z[n-1], \ldots, z[n-M+1]\big\}$, M being larger or equal to the memory length of the nonlinear transformation $\mathbb{N}$, is used. The Newton step in (4.19), whose computation is the main contribution to the complexity of the method, can be approximated by a so-called quasi-Newton step [75], which is the solution of

$$\mathbf{B}_i \delta z_i[n] = -\mathbb{T}_u\big(z_i[n]\big) \ . \tag{4.24}$$

The secant condition requires [75] that

$$\mathbf{B}_{i+1} \delta z_i[n] = \delta \mathbb{T}_u\big(z_i[n]\big) \ , \tag{4.25}$$

with $\delta\mathbb{T}_u\big(z_i[n]\big) = \mathbb{T}_u\big(z_{i+1}[n]\big) - \mathbb{T}_u\big(z_i[n]\big)$. Considering the general case with M dimensions and thus avoiding the time-variant formulation of (4.23), a multidimensional secant method must be used. The problem here is that in more than one dimension the matrix $\mathbf{B}_{i+1}$ in (4.25) cannot be uniquely determined, which gives additional degrees of freedom. The most popular and in practice best performing algorithm is Broyden's method [76, 75]. For the implementation in real-time, where the output of the pre-filter must be calculated on a sample-by-sample basis, the time-variant formulation in (4.23) is adequate. Broyden's method is block-based, and thus not applicable for a real-time implementation.

In one dimension, $\mathbf{B}_{i+1}$ is simply a scalar b_{i+1} and is for the transformation in (4.16)

$$b_{i+1} = \frac{\mathbb{N}\big(z_{i+1}[n]\big) - \mathbb{N}\big(z_i[n]\big)}{z_{i+1}[n] - z_i[n]} \ , \tag{4.26}$$

as can easily be verified. Thus, the secant step from (4.24)

$$\delta z_i[n] = -b_i^{-1}\mathbb{T}_u\big(z_i[n]\big) = -\frac{z_i[n] - z_{i-1}[n]}{\mathbb{N}\big(z_i[n]\big) - \mathbb{N}\big(z_{i-1}[n]\big)} \mathbb{T}_u\big(z_i[n]\big) \tag{4.27}$$

results.

If the algorithm converges, the convergence rate of the secant method is superlinear [75]

$$\big\|z_i[n] - z[n]\big\| \leq c\big\|z_{i-1}[n] - z[n]\big\|^{\phi} \ , \tag{4.28}$$

with the golden ratio $\phi = \frac{1}{2}\big(1+\sqrt{5}\big) \approx 1,618$. The convergence order is smaller than that of the Newton method, but still much larger than the convergence order of the successive approximation method.

4.4 Applying the Linearisation Methods

In Sections 4.4.1 and 4.4.2 the two presented linearisation methods, namely the successive approximation method, discussed in Section 4.2 and the secant method, discussed in Section 4.3, are compared with respect to convergence rate and robustness against model uncertainties. A high convergence rate is essential for a practical real-time implementation of the pre-distortion filter, e.g., in an FPGA or a DSP, since few iterations are required for a good approximation of the ideal pre-filter. Further, in a practical implementation the system (power-amplifier with analogue circuitry such as modulators, demodulators, and mixers) cannot be modelled perfectly. Various types of disturbances affect the modelling:

- At the output of the system, measurement noise affects the estimation of the model parameters. The noise comes from various sources, e.g., due to the limited resolution of the analogue-to-digital converters, quantisation noise is introduced. Other components, such as mixers, demodulators, and amplifiers add thermal noise.
- The model-structure does not equal the system-structure. The power-amplifier is not a specific Volterra-, Wiener-, or Hammerstein system, the model being thus only a more or less accurate approximation to the system.

These disturbances will ultimately limit the achievable linearisation quality, since the design of the pre-filter is based on the model.

The following situations are investigated:

1. Model-parameter errors due to measurement noise
2. Memory effects are neglected in the model
3. Underestimation of the nonlinear order of the system

This investigation is performed using simulations since a direct control of the system output is necessary, e.g., for adding measurement noise. The “test-system” in this section is the Volterra model of the Motorola power-amplifier of Section 2.6. It can be considered as realistic. The different situations are then simulated and their influence on the linearisation quality is investigated.

First, the parameters of the test-system $\mathbb{S}$, see Fig. 4.5, are estimated using the input/output data obtained from the ADS simulation. This test-system replaces the “real”, physical system in the following simulations. Then, noise is added to the output data obtained from the ADS simulation of the power-amplifier. The noise is white, zero-mean and with a gaussian amplitude distribution. The model has the same structure as the test-system, i.e., it is also a Volterra system of order five with kernels of length two. The implication of the added noise is a higher modelling error.

Subsequently, the effect of model-system mismatch, i.e., a structural difference of system and model, is investigated. In this case, no measurement noise is added to the output data obtained by the ADS simulation. The error signal due to imperfect modelling is

$$\tilde{e}_{mod}[n] = \tilde{y}[n] - x[n] = \mathbb{S}\big(z[n]\big) + v[n] - \mathbb{N}\big(z[n]\big) \; . \tag{4.29}$$

The modelling quality is measured by

$$J_{mod}\,[dB] = 10 \log \left(\frac{\big\|\tilde{e}_{mod}[n]\big\|_2^2}{\big\|\tilde{y}[n]\big\|_2^2} \right) \; . \tag{4.30}$$

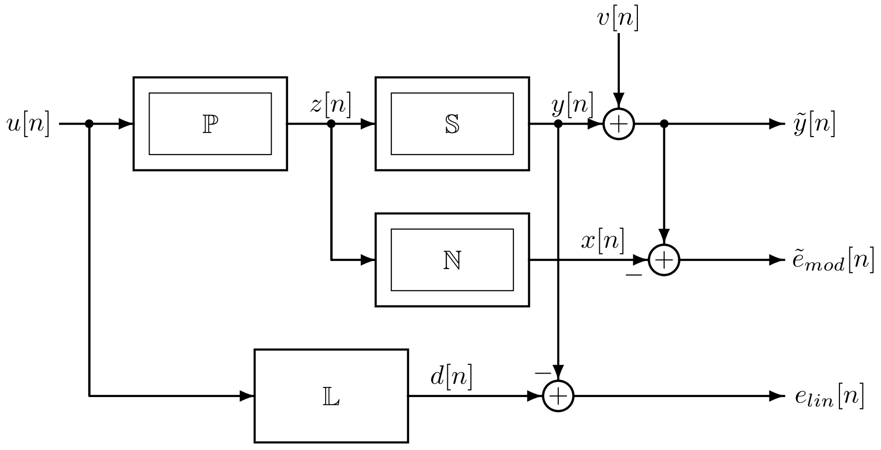


Figure 4.5: Influence of uncertainties on the linearisation – error signals

Since the linearisation algorithms presented in Section 4.2 and Section 4.3 use the model $\mathbb{N}$ to determine the pre-filter $\mathbb{P}$ and thus the output $z[n]$, perfect linearisation of the system $\mathbb{S}$ is in general not possible. The error signal

$$e_{lin}[n] = d[n] - y[n] = \mathbb{L}\big(u[n]\big) - \mathbb{S}\big(z[n]\big) \tag{4.31}$$

will not vanish. The quality of the linearisation due to the pre-filter $\mathbb{P}$ is measured by

$$J_{lin}\,[dB] = 10\log\left(\frac{\|e_{lin}[n]\|_2^2}{\|d[n]\|_2^2}\right)\,, \tag{4.32}$$

in which the undisturbed system output $y[n]$ is used, see Fig. 4.5. Therewith, the effective linearisation of the system is measured.

In Section 4.4.3 measurement results are presented. The high-power EDGE amplifier, modelled in Section 2.6.1, is linearised with the presented secant method [12]. It is shown in an experiment that with the presented digital pre-distortion method, a microwave power-amplifier can be linearised efficiently.

4.4.1 Comparison of the Convergence Rate

The convergence rates of the successive approximation method and the method based on the secant algorithm are compared. For this, the Volterra model of the Motorola LDMOS power-amplifier modelled in Section 2.6.1, cf. also Tab. 2.6, is used as the “system” to linearise. In this case, the model $\mathbb{N}$, which is used by both linearisation algorithms to calculate the pre-distortion filter output $z[n]$, either using (4.11) or (4.23), equals the system. The targeted linear system is just a linear amplification $\mathbb{L}\big(u[n]\big) = g \cdot u[n]$, where the selected linear gain $g = 3,7$, which is approx. the slope of the AM-AM characteristic, cf. Fig. 2.11.

In Fig. 4.6 the linearisation error

$$J_{lin}(i)[dB] = 10\log\left(\frac{\big\|d[n] - y_i[n]\big\|_2^2}{\big\|d[n]\big\|_2^2}\right) \tag{4.33}$$

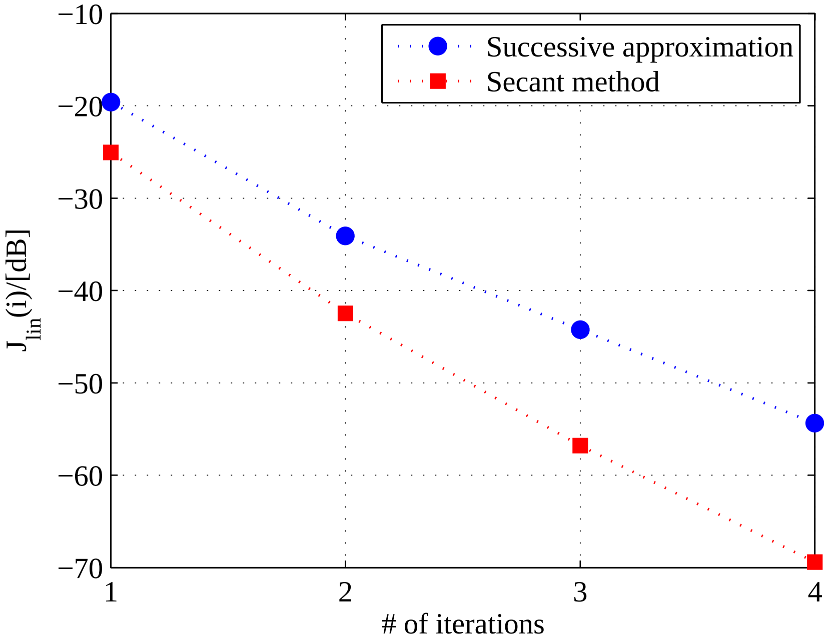


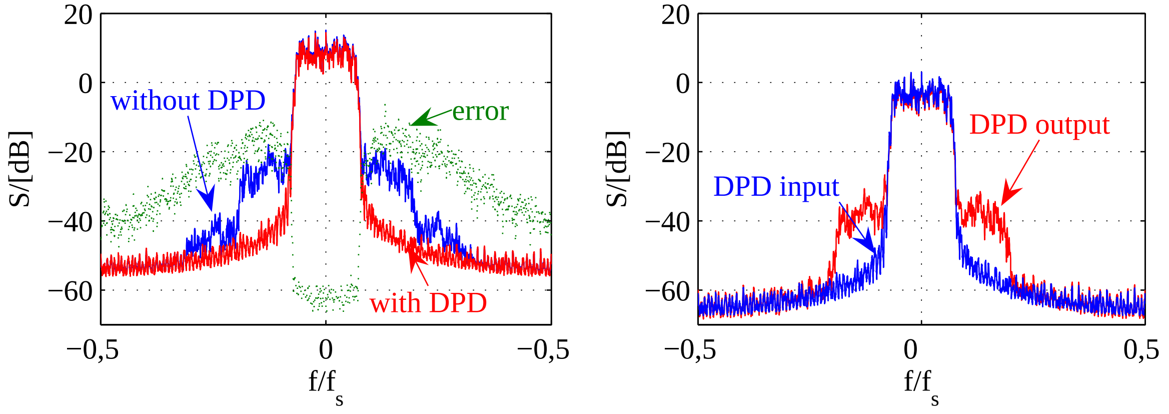
Figure 4.6: Comparison of the convergence speed of the successive approximation method and the method based on the secant algorithm

vs. the number of performed iterations i is shown. Here, $d[n] = g \cdot u[n] = 3,7 \cdot u[n]$ and $y_i[n]$ is the output of the system with the pre-filtered input $z_i[n]$, corresponding to a pre-filter employing i iterations, see (4.12) or (4.23).

With successive approximation the linearisation error decreases approx. with 10 dB per iteration, whereas using the secant method the error-decrease is approx. 16 dB per iteration. This corresponds exactly to the linear convergence rate of the successive approximation method (4.14) and the superlinear convergence rate of the secant method (4.28), respectively. With the secant method, the error after only one iteration is about 6 dB smaller than with successive approximation. The starting values are $z_0[n] = u[n]$ for the successive approximation method and $z_{-1}[n] = 0, z_0[n] = u[n]$ for the secant algorithm. To achieve a linearisation error of, e.g., -56,8 dB, the secant algorithm requires three iterations. With the successive approximation method at least five iterations have to be performed to achieve this value.

In Fig. 4.7(a) the spectrum of the linearised Volterra system using the secant method after three iterations with perfect system knowledge (model equals the system) is shown. It can be seen that the spectral regrowth is perfectly compensated for as expected for a perfect linear amplification. However, a small decrease in the output power of approx. 0,6 dB is noted. This effect results from the fact that the targeted linear gain has to be selected in such a way that the amplifier is still able to deliver the required signal amplitude even in saturation. In part (b) of the Fig. 4.7 the spectra of the input and output signal of the pre-filter $\mathbb{P}$ are shown. The spectral regrowth due to the nonlinear pre-filter is clearly noticeable.

Fig. 4.8 illustrates the AM-AM and AM-PM conversion of the Volterra system with and without linearisation. Part (a) shows the AM-AM conversion. The magnitude of the system output is $|y[n]|$, the magnitude of the digital pre-distortion filter output (DPD-output) is $|z[n]|$, and the magnitude of the input signal is $|u[n]|$. Practically perfect linearisation is

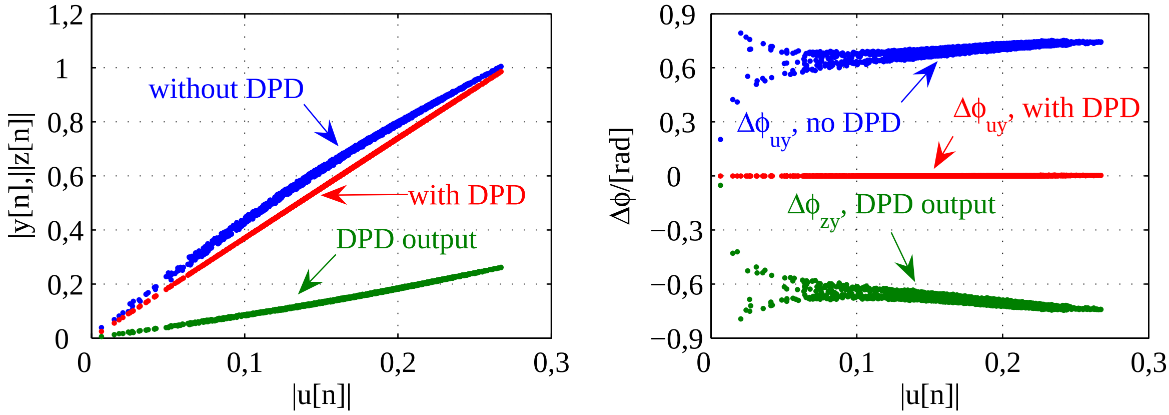


(a) System output with/without linearisation

(b) Spectra of the input and output signal of the pre-filter $\mathbb{P}$

Figure 4.7: Spectra of input and output signals of the Volterra system $\mathbb{S}$ (with and without digital pre-distortion (DPD)) and the pre-filter $\mathbb{P}$

achieved. Part (b) shows that the AM-PM conversion is also compensated perfectly. The difference between the phase of the system output and the phase of the input signal $\Delta\Phi_{uy}[n] = \arg\big(y[n]\big) - \arg\big(u[n]\big)$ and the difference between the the pre-distortion filter output and the input signal $\Delta\Phi_{uz}[n] = \arg\big(z[n]\big) - \arg\big(u[n]\big)$ are shown. The overall achieved linearisation error, defined in (4.32), is $J_{lin} = -56,8\,\text{dB}$, see also Fig. 4.6.



(a) AM-AM conversion

(b) AM-PM conversion

Figure 4.8: AM-AM and AM-PM conversion before and after linearisation

4.4.2 System-Model Mismatches

Influence of Measurement Noise

For the estimation of the model-parameters the output data obtained from the ADS simulation is disturbed with noise. The amount of disturbance is measured using the signal-to-noise ratio

$$\mathrm{SNR}\,[dB] = 10 \log \left(\frac{\|y[n]\|_2^2}{\|v[n]\|_2^2} \right) . \tag{4.34}$$

Fig. 4.9 shows the linearisation error of the successive approximation method if the output signal of the system, cf. Fig. 4.5, is affected by white, zero-mean, gaussian noise $v[n]$. The results are obtained by taking the average of the linearisation error of 20 different simulation runs with different noise realisations. The error shows little dependence on the SNR at the system output if less then five iterations are performed. The convergence rate of the successive approximation method is too small – the minimum linearisation error cannot be achieved. If more iterations are performed, the linearisation error decreases with increasing SNR. In the considered SNR interval of 50 dB – 65 dB, with seven iterations the lower limit of the linearisation error can be reached.

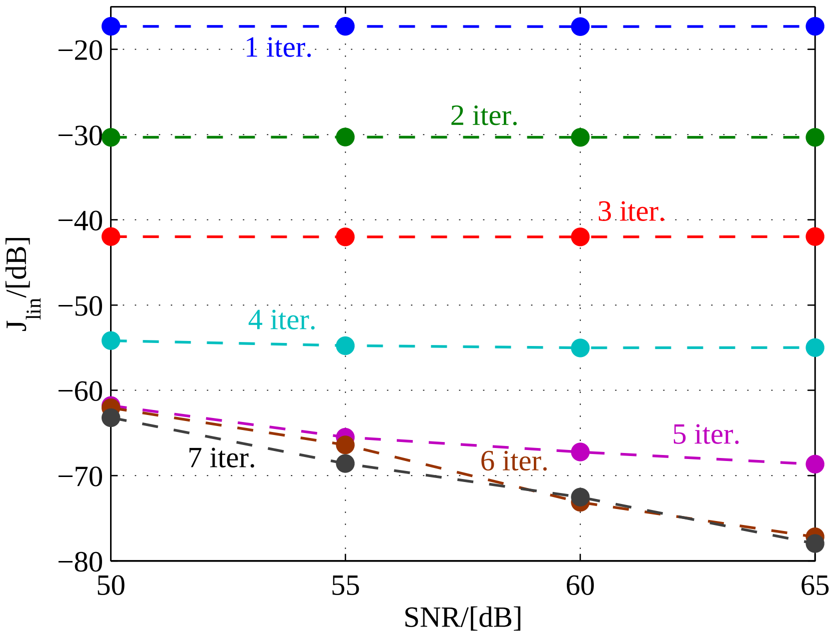


Figure 4.9: Influence of measurement noise on the achieved linearisation error J_{lin} – successive approximation method

Fig. 4.10 illustrates the linearisation error of the secant method vs. the SNR at the system output. Here, the convergence speed is high enough to achieve the minimum possible linearisation error with only four iterations. In the considered SNR range no further improvement can be achieved if five iterations are performed. If four iterations are performed instead of only three, the reduction of the linearisation error at an SNR = 55 dB is only a few dB. If three iterations are performed instead of only two, the reduction of the linearisation error is much larger, approx. 16 dB. For a practical dynamic range of 50–60 dB three iterations with the secant method are therefore sufficient.

The reduction of the number of iterations is essential since it reduces the hardware complexity (e.g., invested number of gates in an FPGA) significantly. If the successive approximation method is used, at least one or two additional iterations have to be performed to achieve the same linearisation error as with the secant method. This results in a significant increase of the hardware complexity.

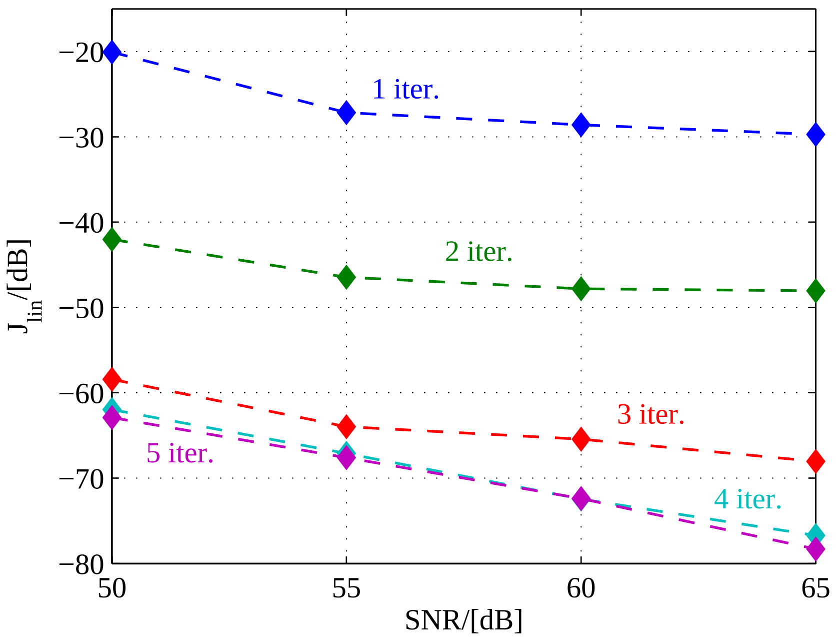
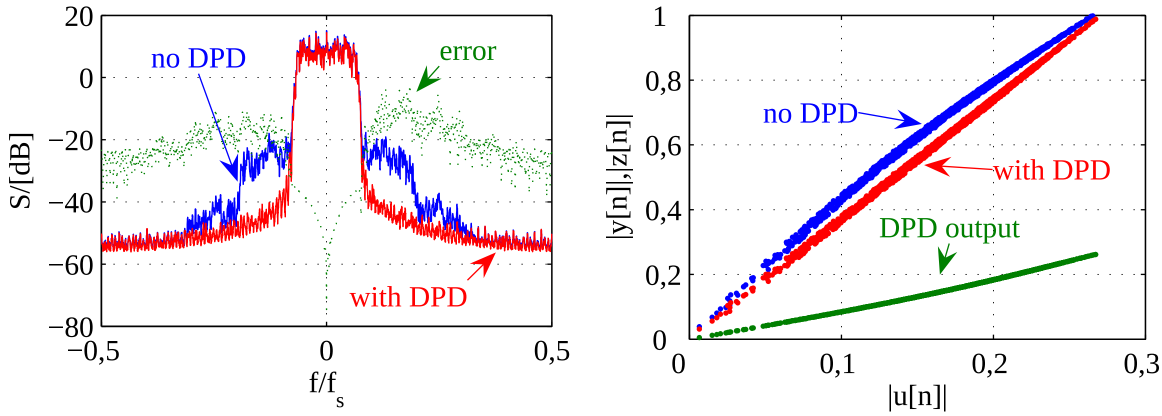


Figure 4.10: Influence of measurement noise on the achieved linearisation error J_{lin} – secant method

Neglecting Memory Effects

In this section the model is a static nonlinear function of nonlinear order five. Consequently, the pre-filter can correct only the nonlinear distortions but not the dispersive effects. The spectrum shows therefore no spectral regrowth, cf. Fig. 4.11(a). The dispersive effects are not corrected as can be recognised form Fig. 4.11(b). The AM-AM characteristic with the pre-filter shows the characteristic broadening due to the not compensated dispersive effects.

In Tab. 4.1 the resulting linearisation errors are reported. With both methods, successive approximation and secant algorithm, three iterations are performed. The loss in linearisation accuracy due to the static modelling $\Delta J_{lin}\,[\text{dB}] = J_{lin,\text{static}}\,[\text{dB}] - J_{lin,\text{Volterra}}\,[\text{dB}]$ is considerable, in particular if the secant method is used. Here, $J_{lin,\text{Volterra}}$ is the linearisation error if the Volterra model is used to design the pre-filter and $J_{lin,\text{static}}$ is the linearisation error if the static model is used for designing the pre-filter. Even if seven iterations are performed the linearisation error can not be reduced further as Tab. 4.2 reports. Both methods achieve the lower limit of the linearisation error of $J_{lin} = -37,8\,\text{dB}$. This shows that with the secant method the best possible linearisation can be achieved with only three iterations, whereas with the successive approximation method more iterations are necessary.



(a) Spectra of system output with and without pre-filter

(b) AM-AM conversion with and without pre-filter

Figure 4.11: Output signals with and without pre-filter – memory effects are not considered, $f_s = 8 \cdot 3,84\,\mathrm{MHz} = 30,72\,\mathrm{MHz}$.

model	J_{lin} [dB], succ. approx.	J_{lin} [dB], secant meth.
Volterra, [2,2,2]	-44,2	-56,8
Static, [1,1,1]	-37,1	-37,8
ΔJ_{lin} [dB]	7,1	19

Table 4.1: Linearisation errors if memory effects are neglected. Three iterations are performed.

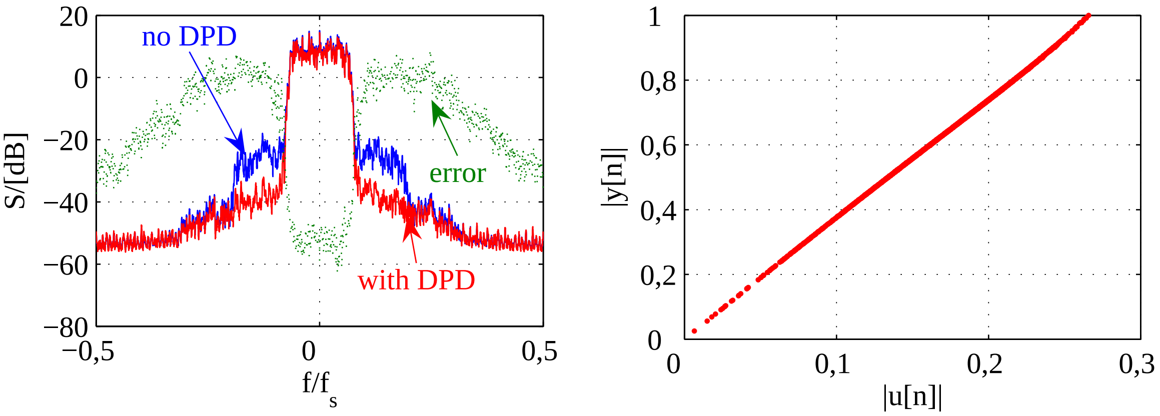
model	J_{lin} [dB], succ. approx.	J_{lin} [dB], secant meth.
Volterra, [2,2,2]	-90,3	-90,3
Static, [1,1,1]	-37,8	-37,8
ΔJ_{lin} [dB]	52,4	52,4

Table 4.2: Linearisation errors if memory effects are neglected. Seven iterations are performed.

Underestimating the Nonlinear Order

Assuming a Volterra model of nonlinear order three, each kernel has two taps, and the system being a Volterra system of order five with each kernel having also two taps, will not result in a perfect compensation of the nonlinear distortions. The third-order distortion will be compensated with the pre-filter, the distortion due to the fifth order terms can obviously not be compensated. Fig. 4.12 visualises the situation: In part (a) the spectrum of the system output signal is shown with and without pre-filter. The spectral regrowth is not compensated perfectly. Part (b), where the resulting AM-AM conversion with pre-filter is shown, the remaining small nonlinear behaviour is also visible. The dispersion is compensated only partially. The dispersion introduced by the fifth order Volterra kernel of the system is obviously not compensated.

In Tab. 4.3 the resulting linearisation errors are listed, together with the loss in performance $\Delta J_{lin}\,[\mathrm{dB}] = J_{lin,[2,2]}\,[\mathrm{dB}] - J_{lin,[2,2,2]}\,[\mathrm{dB}]$ due to the system-model mismatch. The loss is not as high as in the previous case, where the memory effects were neglected, but is still considerable.



(a) Spectra of system output with and without pre-filter

(b) AM-AM conversion with pre-filter

Figure 4.12: Output signals with and without pre-filter – the system is of order five, the model is of order three, $f_s = 8 \cdot 3,84\,\text{MHz} = 30,72\,\text{MHz}$.

If seven iterations are performed the linearisation error can not be reduced further as Tab. 4.4 reports. Also in this setting, both methods achieve the lower limit of the linearisation error of $J_{lin} = -43,6\,\text{dB}$. With the secant method the best possible linearisation can be achieved with three iterations, whereas with the successive approximation method more iterations are necessary.

model	J_{lin} [dB], succ. approx.	J_{lin} [dB], secant meth.
Volterra, [2,2,2]	-44,2	-56,8
Volterra, [2,2]	-41,7	-43,6
ΔJ_{lin} [dB]	2,5	13,2

Table 4.3: Linearisation errors if the nonlinear order is underestimated. Three iterations are performed.

model	J_{lin} [dB], succ. approx.	J_{lin} [dB], secant meth.
Volterra, [2,2,2]	-90,3	-90,3
Volterra, [2,2]	-43,6	-43,6
ΔJ_{lin} [dB]	46,7	46,7

Table 4.4: Linearisation errors if the nonlinear order is underestimated. Seven iterations are performed.

4.4.3 Linearising a High-Power LDMOS EDGE Amplifier – Measurement Results

In this section a high-power LDMOS class AB microwave amplifier with three stages for application in GSM-EDGE is linearised [12]. The secant method is used to calculate the pre-distorted signal. Three iterations are performed.

The measurement setup is shown in Fig. 4.13. The test-signals are generated in the PC. The signal for modelling is a multi-tone signal with a bandwidth of 1 MHz and 101 tones. The tones are equally spaced and have random phases, drawn from a uniform distribution over the interval $[0, 2\pi)$. The crest-factor of this signal is with 8,5 dB relatively high.

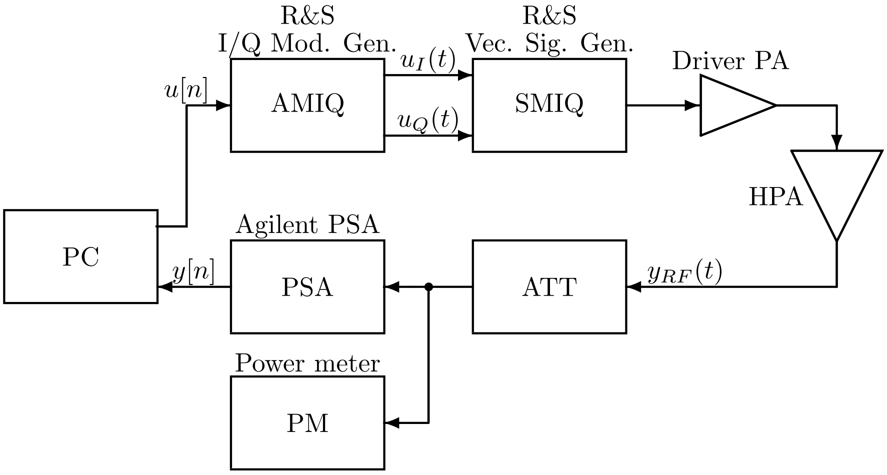


Figure 4.13: Measurement setup – linearisation of the high-power EDGE amplifier

The complex digital signal $u[n]$, created in the PC, is converted to two analogue signals $u_I(t)$ and $u_Q(t)$ using the Rohde & Schwarz I/Q Modulation Generator AMIQ. The resolution of the digital-to-analogue converters is 14 bits. In combination with the Rohde & Schwarz Vector Signal Generator SMIQ it is possible to create a modulated carrier at a selectable carrier-frequency, which in this case is $f_c = 1,9$ GHz, corresponding the the GSM-EDGE specifications. A single-stage driver amplifier (Mini-Circuits ZHL-42W [58]) boosts the signal at a sufficiently high power level for the high-power amplifier (HPA). The driver amplifier has a minimum gain of 30 dB and is a highly linear class A measurement amplifier. If it is driven well below the 1 dB compression-point, the only source of nonlinear distortions is the high-power EDGE amplifier.

The Vector Signal Generator allows to vary the power level; with this, the saturation and the total output power of the EDGE high-power amplifier can be controlled. Different levels of input-power (corresponding to various levels of input back-off, IBO) can thus be delivered to the amplifier chain. After attenuation of the output signal of the high-power amplifier, the signal is fed to a signal analyser (Agilent PSA) and to a power-meter. The power-meter is used to control precisely the total output power. This is necessary since for a correct evaluation of the digital pre-distortion the output power with and without pre-distortion must be guaranteed to be the same. Since the signal analyser allows only for accurate relative power measurements the power-meter is needed.

The signal analyser down-converts and demodulates the signal to baseband. The signal analyser is synchronised with the SMIQ Vector Signal Generator to guarantee equal phases of the internal local oscillators for the I/Q modulation- and demodulation. The baseband signal is then converted to digital and can be stored on a PC. The signal analyser has an analysis bandwidth of 8 MHz and uses a sampling frequency of $f_{s,\mathrm{PSA}} = 10,24$ MHz. Consequently, applying an input signal with a bandwidth of 1 MHz, harmonic out-of-band distortions up to the seventh order are completely included in the recorded signal.

The procedure is now as follows:

1. First, the multi-tone test signal is applied to the power-amplifier chain. Here, three different levels of input power, corresponding to different levels of saturation, are used.
2. With the input-signal and the recorded output-signal, a black-box model of the amplifier is created. A Volterra model and a Wiener model are compared for all three levels of saturation.
3. Since, as is shown in the following, the Wiener model results in a comparable modelling error, it is used to calculate the pre-distorted signal with the secant method.
4. The pre-distorted signal is then used to excite the power amplifier chain. The output signal is stored in the PC. The power level is measured with the power-meter.
5. Since the power level with pre-distortion is smaller than in the first step, the undistorted signal has to be applied once more. The output power of the SMIQ is slightly reduced, compared to step one, in order to guarantee the same output power of the amplifier chain as with pre-distortion. This corresponds to an input back-off.

Modelling

Two models are compared using three different degrees of saturation of the high-power amplifier: A Volterra model and a Wiener model. The amplifier input power is adjusted to different levels with the SMIQ. The models are then fitted to the measured data. Both models have a nonlinear order of seven. For the Volterra model, the least modelling error was found with a linear kernel of length five, a third-order kernel with three taps, a fifth-order kernel with two taps and a seventh-order kernel with one tap (memoryless). The corresponding structural notation is $m_V = [5, 3, 2, 1]^T$. The best Wiener model is a four tap FIR filter followed by a seventh order static non-linearity. In Tab. 4.5 the three different cases, corresponding to the three different power levels, are compared. The modelling error, defined in (4.30), of the Wiener model is approx. 1,6 dB higher as the modelling error of the Volterra model. The advantage of the Wiener model is the small number of parameters: only eight complex parameters are required whereas the Volterra model requires 36 parameters. The modelling quality decreases further with increasing output power. This is reasonable, since with increasing output power the high-power amplifier is driven more and more into saturation, corresponding to a relatively hard non-linearity, which is difficult to fit with only a seventh order model. On the other hand, out-of-band distortions higher than seventh-order cannot be analysed by the signal analyser due to the limited bandwidth of 8 MHz. Only the weak in-band distortions are available for the modelling, making thus the task of fitting higher nonlinear terms difficult. Therefore, the nonlinear model-order is kept constant, even if the amplifier is driven in higher saturation.

case	P_{out}/[dBm]	$J_{\text{mod,Wiener}}$/[dB]	$J_{\text{mod,Volterra}}$/[dB]
1	42,3	-37,3	-38,9
2	43,9	-35,5	-37,1
3	45,4	-33,8	-35,3

Table 4.5: Modelling errors – high-power EDGE amplifier

Fig. 4.14 shows the spectrum of the measured output signal of the EDGE-HPA and the spectrum of the Wiener model for case 1, see Tab. 4.5. The in-band signal and the out-of-band third order harmonic distortions are approximated very accurately. The very small ($< -50\,\mathrm{dBc}$) fifth and higher order out-of-band distortions are not well approximated. The estimation of these intermodulation products is affected by the limited dynamic range (which is approx. 60 dB) of the equipment and the resulting low signal-to-interference ratio. In this case of a moderate saturation-level the third-order distortion is clearly dominating. It has to be noted that also the higher order nonlinear terms are contributing to the distortion in the third-order out-of-band zone. Using a seventh-order model improves the modelling, although the distortion in the corresponding out-of-band harmonic zone is very small.

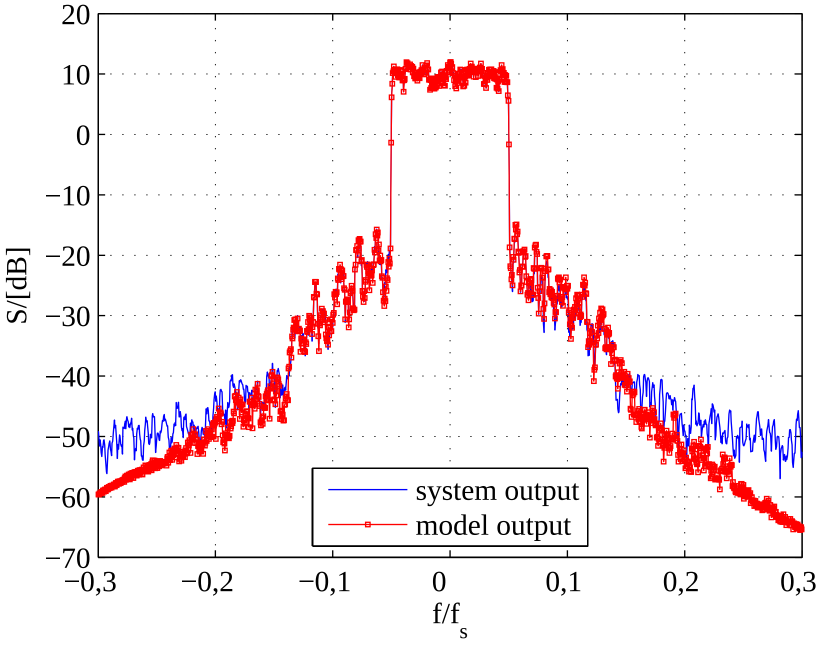


Figure 4.14: Output spectra of the EDGE-HPA and the corresponding Wiener model, $f_s = 10,24\,\mathrm{MHz}$

Linearisation

After modelling, the signal at the output of the pre-distortion filter is computed. For this, the secant method, described in Section 4.3, is applied. In Sections 4.4.1 and 4.4.2 simulations demonstrated that with a dynamic range of 50-60 dB three iterations are optimal. More iterations improve the linearisation only marginally and increase the complexity unnecessarily, see, e.g., Fig. 4.10.

Once the signal after the pre-filter is determined it excites the power amplifier chain. The (small) nonlinear distortion of the original multi-tone signal, introduced by the nonlinear pre-distortion, implicate a change in the input power and thus also a change in the output power of the high-power amplifier. In general, the power level with pre-distortion is smaller than the power level without pre-distortion if the amplifier operates in saturation, see Fig. 4.7. This is due to the fact that only a smaller linear gain guarantees that a solution of (4.16) exists for the whole input range. Therefore, for a correct comparison, the original undistorted input signal has to be applied to the amplifier-chain once again and the input power level has to be adjusted in such a way that the output power equals the output power with pre-distortion. This corresponds to an output-power back-off (OBO) and yields thus a more linear behaviour of the amplifier. The question is whether pre-distortion reduces the nonlinear distortions more than the simple back-off.

Fig. 4.15 shows the spectra of the measured output signal with and without pre-distortion. The signals are normalised, for the exact power level cf. Tab. 4.6. The nonlinear distortions are significantly reduced, with pre-distortion approx. -45 dBc are achieved, with back-off only -30 dBc are obtained in the immediate neighbourhood of the in-band region. Notice that for this case, an output back-off of 1,4 dB does not result in a significant decrease of the out-of-band distortions as can be seen from Fig. 4.14 and Fig. 4.15. The improvement in terms of reduction of the out-of-band distortion is significantly higher if pre-distortion is used.

Fig. 4.16 shows a detailed view of the third-order out-of-band zones, extending over the normalised frequency intervals $f/f_s = [-0,15; -0,05]$ and $f/f_s = [0,05; 0,15]$. The performance gain by using pre-distortion vs. back-off is clearly visible. With increasing distance to the in-band region, the improvement decreases, according to the decrease of the nonlinear distortions.

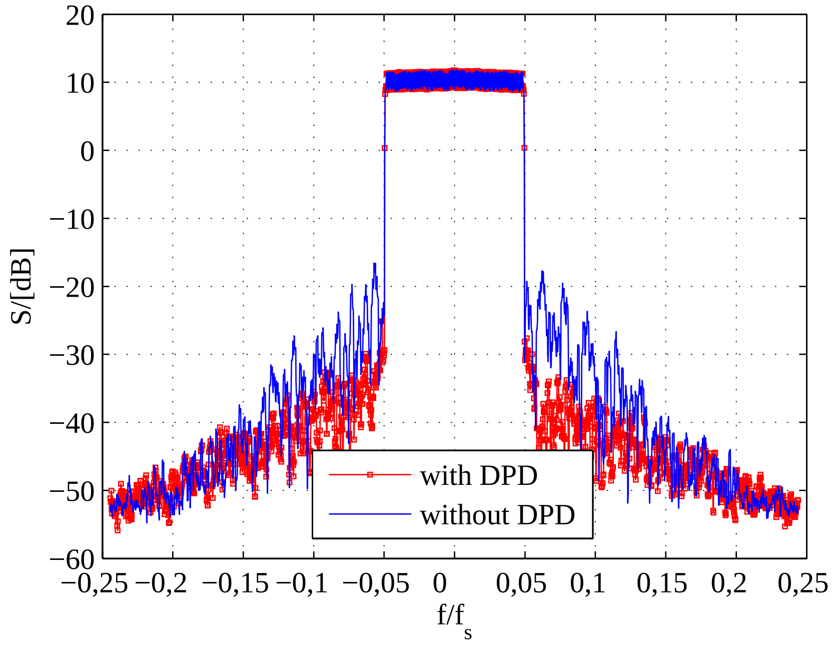


Figure 4.15: Spectra of the measured output signal with and without digital pre-distortion, $f_s = 10,24\,\mathrm{MHz}$

In order to quantify the performance gain, the total power in the third-order out-of-band zone using the back-off approach and the pre-distortion approach is accumulated. For this, the spectral power density in the (normalised) frequency range $I_{L,3} = [-0,15; -0,05]$ corresponding to the lower third-order zone and the frequency range $I_{U,3} = [0,05; 0,15]$ corresponding to the upper third-order zone is accumulated. The measure, which quantifies the linearisation-

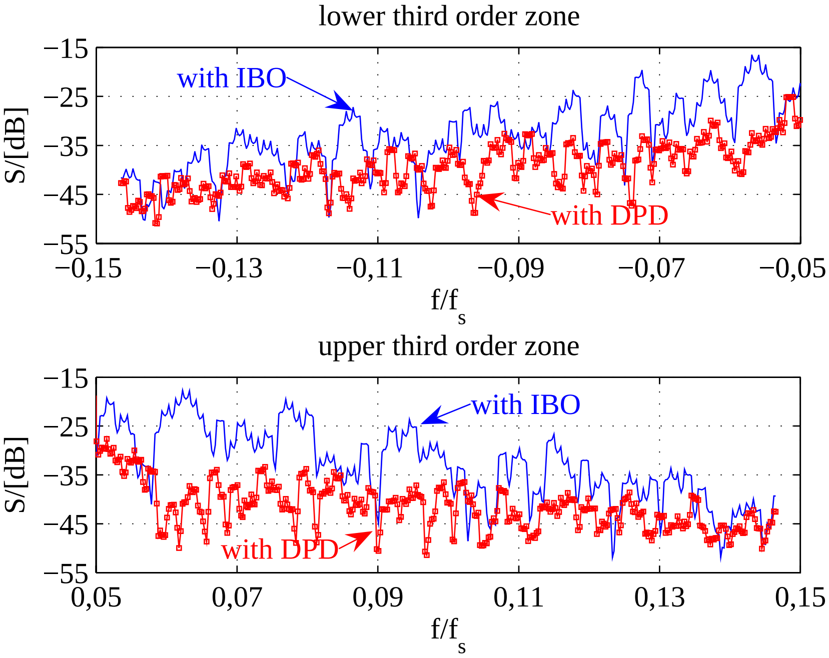


Figure 4.16: Spectra of the measured output signal with and without digital pre-distortion – detailed view of the upper and lower out-of-band third-order harmonic zone, $f_s = 10,24\,\mathrm{MHz}$

gain of the pre-distortion vs. back-off is

$$g_{\mathrm{Lin},3}[\mathrm{dB}] = 10 \log \left(\frac{P_{3,\mathrm{OBO}}}{P_{3,\mathrm{DPD}}} \right) . \tag{4.35}$$

Here, $P_{3,\mathrm{OBO}}$ is the power in the third-order out-of-band zone with output back-off, whereas $P_{3,\mathrm{DPD}}$ is the power in this zone if digital pre-distortion (DPD) is used. In Tab. 4.6 the so defined pre-distortion gain is listed for all three cases. With increasing saturation (higher power level) the gain decreases – higher order nonlinear products become more pronounced but cannot be reduced due to the inaccurate modelling, caused by the limited bandwidth and the limited dynamic range of the measurement equipment.

The overall deviation from the targeted linear amplification is quantified by the distance $\|z[n] - d[n]\|_2$, where $d[n] = g \cdot u[n]$ is the targeted linear amplification of the input signal and $z[n]$ is the measured output signal, either with back-off or with pre-distortion. The linearisation-gain using pre-distortion compared to back-off can thus be quantified by

$$g_{\mathrm{Lin}}[\mathrm{dB}] = 20 \log \left(\frac{\|z_{\mathrm{OBO}}[n] - d[n]\|_2}{\|z_{\mathrm{DPD}}[n] - d[n]\|_2} \right) , \tag{4.36}$$

where $z_{\mathrm{OBO}}[n]$ is the measured amplifier output signal with back-off, and $z_{\mathrm{DPD}}[n]$ is the measured output signal with pre-distortion. In case 1, which corresponds to the least output power and therefore to the case where the power amplifier is not too much in saturation, pre-distortion yields a significant gain compared to the simple back-off. The out-of-band power in the third-order zone was reduced significantly, as well as the total distortion. Driving the power amplifier with a higher input power reduces the gain achieved with the pre-distortion, compared to the back-off approach, but it is still advantageous to use pre-distortion. The required back-off is larger in these cases, which corresponds to a larger gain-decrease when pre-distortion is used.

case	P_{out}/[dBm]	OBO/[dB]	g_{Lin}/[dB]	$g_{\text{Lin},3}$/[dB]
1	40,9	1,4	9,7	10,7
2	41,6	2,3	11,1	6,7
3	43,3	2,1	3,3	4,1

Table 4.6: Reduction of out-of-band power in the third-order harmonic zone using pre-distortion vs. back-off ($g_{\text{Lin},3}$) and reduction of overall distortion g_{Lin}

4.5 Discussion

Two approaches for linearisation of a nonlinear dynamic system were presented. The first approach is based on a fixed-point formulation of the pre-distortion problem and applies the successive approximation method for its solution [18]. The second approach solves the pre-distortion problem via a root-search applying the secant method [12]. Hence, both approaches solve the problem in an approximate way using an iterative technique. Further, these approaches do not assume a specific nonlinear pre-filter, since for most nonlinear systems the structure of the linearising pre-filter cannot be determined analytically, even for static nonlinearities of moderate orders. The methods determine therefore not a set of optimal parameters for a given pre-filter, but the signal after the pre-filter. Insofar, the iterative procedures, the successive approximation method and the secant method can be seen as the approximation to an ideal linearising pre-filter.

The convergence order of the iterative methods is of significant importance for a real-time implementation of the pre-filter since a high convergence order implies either few implemented stages (one stage corresponds to one iteration) or few iterations, if only one linearisation-stage is implemented. The convergence order of the successive approximation method is only linear, whereas the secant method converges with a superlinear order. This comes with a higher complexity: a division has to be performed, which is a rather high-complex operation if it must be implemented in hardware, e.g., in an FPGA. But it can be achieved with high accuracy and a reasonable complexity, as will be shown in Section 5.4.1 in Chapter 5.

A further disadvantage of both methods is the lack of a convergence test which can be performed with reasonable effort. In the case of the successive approximation method, the convergence analysis can be performed but delivers very conservative results due the the nature of the proof, which gives sufficient conditions for convergence, and uses various approximations (upper bounds for operator norms). In the case of the secant method, no feasible method to prove convergence is known. Due to the fact that the nonlinear power amplifier is in general only weakly nonlinear, both methods do not experience unstable behaviour. This was shown using simulations based on a realistic power amplifier model.

The methods were evaluated by simulations with respect to robustness against modelling errors. Both methods proved to be resistant in the investigated cases. The secant method is, due to its higher convergence order, advantageous, since in general one to two iterations less than with the successive approximation method are required to achieve the same linearisation accuracy.

In an experimental setup, a high-power class AB LDMOS amplifier for the GSM-EDGE standard was linearised with the secant method. It was shown that the proposed method achieves also very good results in an experimental setup and delivers significantly better results than the usual back-off approach.

Chapter 5

Prototype Implementation

This chapter presents a prototype implementation of the pre-distortion algorithm discussed in Chapter 4. Here, a memoryless power amplifier model with only AM/AM conversion is considered in order to keep the implementation-complexity very low. The aim is to have a flexible and extendable first implementation of the presented linearisation algorithm running in real-time in hardware, concretely on an FPGA. Although the developed pre-distortion method itself proved to be capable of linearising a high-power microwave amplifier, as shown in Section 4.4.3, an implementation is necessary to prove that the algorithm can also be realised using a fixed-point environment with a limited amount of resources. These resource-limitations can be of different nature:

- maximal speed, e.g., maximal clock-frequency of the FPGA,
- limited resolution, e.g., 12 bit AD/DA converters, fixed-point arithmetic in the FPGA,
- limited number of arithmetic and/or logic units in the FPGA,
- maximal bus data-rates, and
- maximal allowed power consumption.

If the algorithm can be implemented and proves to be working, further optimisations and/or extensions are to be performed to satisfy product constraints.

In a prototyping implementation several aspects regarding the future product realisation can be neglected, e.g., the amount of hardware resources invested. The first principal aim is to prove the feasibility of the algorithm, without placing the emphasis on implementation efficiency. If the principal operation in hardware has been shown, the design can be extended and optimised to satisfy the constraints of a future product. The implementation of a prototype helps therefore not only to understand and possibly solve implementation problems, but to prevent a future production from unexpected difficulties and possibly non-realisable parts of the algorithm. Further, costly parts, as well as parts which are difficult to realise due to high quality constraints, are detected.

In the following, an overview of the realised pre-distortion prototype is presented. After the system concept has been introduced, the required hardware-components are briefly discussed. The realised implementation is then described in a more detailed fashion, before measurement results from this real-time implementation are presented and conclude this chapter.

5.1 System Concept

The system concept of the digital signal processing part is illustrated in Fig. 5.1. The hardware details are presented in Section 5.2.

The digital pre-distortion system is divided into two parts: The digital pre-distortion filter is required to run in real-time. It is therefore implemented in an FPGA. The identification part, responsible for estimating the model parameters, runs in a block-based mode, i.e., the model-parameters are not updated continuously using every incoming data-sample, but using stored data-blocks. This part is implemented in software on the PC. The signal to be transmitted is stored in a memory section and accessible by the identification algorithm. The output signal of the RF-part is sampled by the AD-converter and directly stored in a memory section. This stored output signal, together with the input signal, is then used to estimate the parameters of the power amplifier model. Once the model-parameters are determined, they are transferred directly to the FPGA and determine the pre-distortion filter.

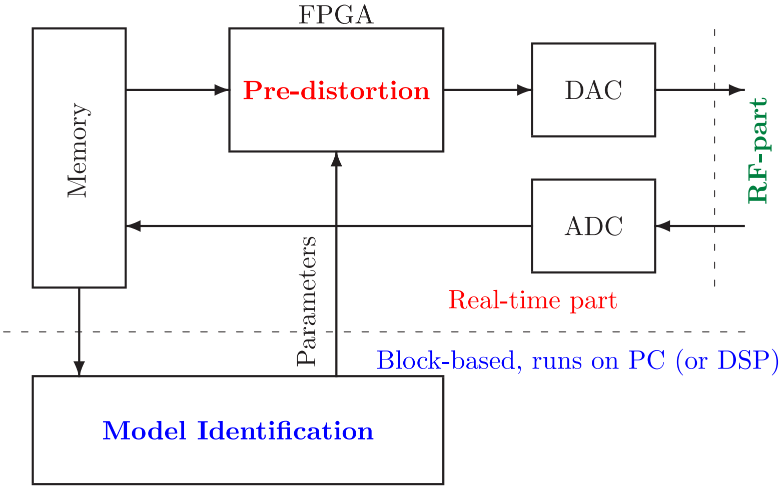


Figure 5.1: Prototype system concept: digital part

The RF-part of the prototype is shown in Fig. 5.2. The DA-converter output signal is first attenuated in order to drive the up-converter with a sufficiently small power, guaranteeing that the up-converter behaves highly linear. After frequency-translation, the analogue RF-signal is amplified with a driver amplifier with fixed gain. In order to have a variable signal-power for the power amplifier targeted to be linearised, a variable attenuator is placed between the driver- and the power amplifier. The output-signal of the power amplifier is attenuated and frequency-translated with the down-converter. Up- and down-converter use both the same local oscillator. The output-signal of the down-converter must be amplified before AD-conversion in order to exploit the whole dynamic range of this converter. For this, two low-noise fixed-gain amplifiers with a variable attenuator are used (not shown in the illustration).

5.2 Signal Processing Hardware – Description

Before the implemented design is presented in more detail the signal processing hardware and the interface to the PC are briefly described in order to introduce the basic features,

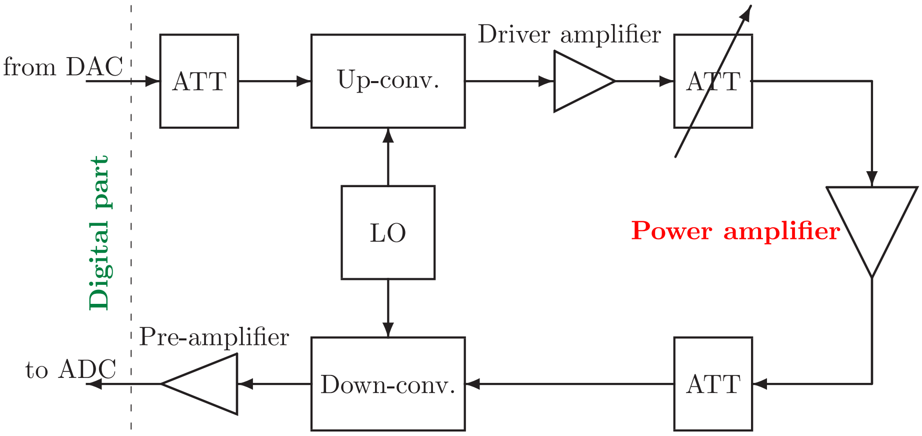


Figure 5.2: Prototype system concept: analogue RF part

terminology, available resources, and, consequently, hardware constraints.

The available hardware consists of:

- an FPGA,
- a DSP,
- a digital-to-analogue converter (DAC),
- a digital up-conversion stage (DUC),
- an analogue-to-digital converter (ADC),
- and a memory block.

In the following each element is introduced briefly, along with its key performance figures.

FPGA The Field Programmable Gate Array (FPGA) is intended to host the pre-distortion filter. It is a XILINX Virtex-II FPGA XC2V1000 [77], a user programmable gate array with various configurable elements. The Virtex-II architecture is optimised for high-density and high-performance logic designs. The Fig. 5.3 shows an architecture overview.

Four major elements are organised in a regular array:

- **Configurable Logic Blocks (CLBs)** provide functional elements for combinatorial and synchronous logic, including basic storage elements. CLBs include four slices and two 3-state buffers. Each slice is equivalent and is comprised principally of two function generators, which are configurable as 4-input look-up tables (LUTs), as 16 bit shift-registers, or as 16 bit distributed SelectRAM memory, two storage elements, arithmetic logic gates, and large multiplexers.
- **Block SelectRAM** memory modules provide large 18 kbit storage elements of dual-port RAM, which are programmable from 16 kbit × 1 bit to 512 bits × 36 bits, in various depth and width configurations.

- **Multiplier blocks** are 18 bit×18 bit dedicated multipliers. Each multiplier block is associated with each SelectRAM memory block. The multipliers are optimised for operations based on the block SelectRAM on one port, but can also be used independently of the block SelectRAM resource. Read/multiply/accumulate operations are extremely efficient.
- **Digital Clock Manager (DCM)** blocks provide fully digital solutions for clock distributions, delay compensation, and clock multiplication and division.

The input/output blocks (IOB) provide the interface to the outside. They are programmable and can be categorised into input blocks with an optional register, output blocks with an optional register and an optional 3-state buffer, and a bidirectional block.

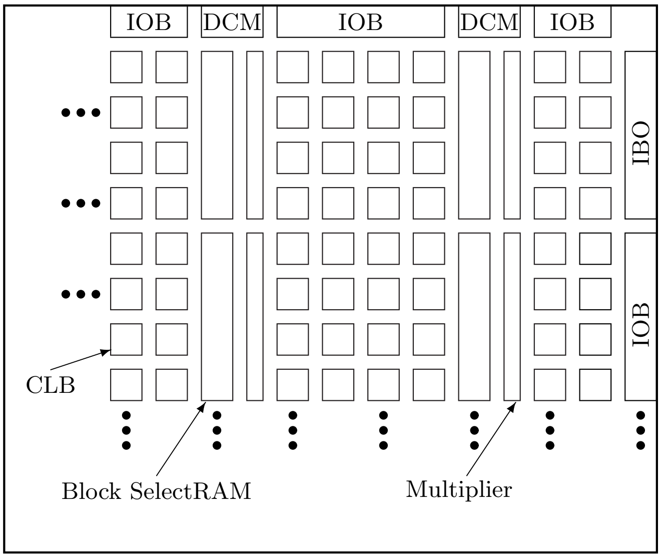


Figure 5.3: Virtex-II architecture overview

Tab. 5.1 presents the amount of resources of the used FPGA which are available for custom designs.

System gates	CLBs		Mult. blocks	SelectRAM Blocks		DCMs
	# of Slices	Max. distrib. RAM		18 kbit blocks	Max. RAM	
1 M	5.120	160 kbit	40	40	720 kbit	8

Table 5.1: Resources of the XILINX Virtex-II XC2V1000 FPGA

DSP The Digital Signal Processor (DSP) used in the realised prototype-design is used as a control device controlling the ADC, DAC, and for setting the model-parameters of the pre-distortion filter running in real-time on the FPGA. In further design-extensions, the DSP is targeted to perform the model-parameter estimation, e.g., using an adaptive algorithm, discussed in Chapter 3. In Tab. 5.2 the key features are summarised [78].

Arithmetic	Cycle-time	# of 32 bit instr./cycle	operations/cycle	# of 32 bit registers
fixed-point	1,67 ns	8	28	64

Table 5.2: Key performance figures of the DSP (TI TMS320C6416)

DUC, DAC Fig. 5.4 illustrates the digital up-conversion (DUC) and digital-to-analogue conversion (DAC) stage, integrated in the dual DA-converter AD9777 [79]. The digital in-phase (I) - and quadrature-phase (Q) input data (with a maximal resolution of 16 bit) is delivered by the FPGA, concretely it is the output of the implemented pre-distortion filter. After interpolation the digital samples are digitally up-converted (DUC), whereby the only possible centre-frequencies f_m are $f_s/2$, $f_s/4$, or $f_s/8$, f_s being the sampling rate of the DAC. The clock is retrieved from the FPGA and a phase-locked loop (PLL) is used to multiply or divide the delivered clock. The DUC, a complex modulator, has two outputs: IF and $\overline{\text{IF}}$. If an analogue quadrature modulator is used for frequency-translation to the targeted centre-frequency $f_c + f_m$, the output $\overline{\text{IF}}$, centred at f_m, is used as the imaginary input-part for the analogue quadrature-modulator. The lower frequency image at $f_c - f_m$ can be suppressed with this method. In the presented prototype-design, the DUC is used for digital modulation to $f_c = 70$ MHz, the frequency-translation is performed with a conventional mixer. The output $\overline{\text{IF}}$ is therefore not used.

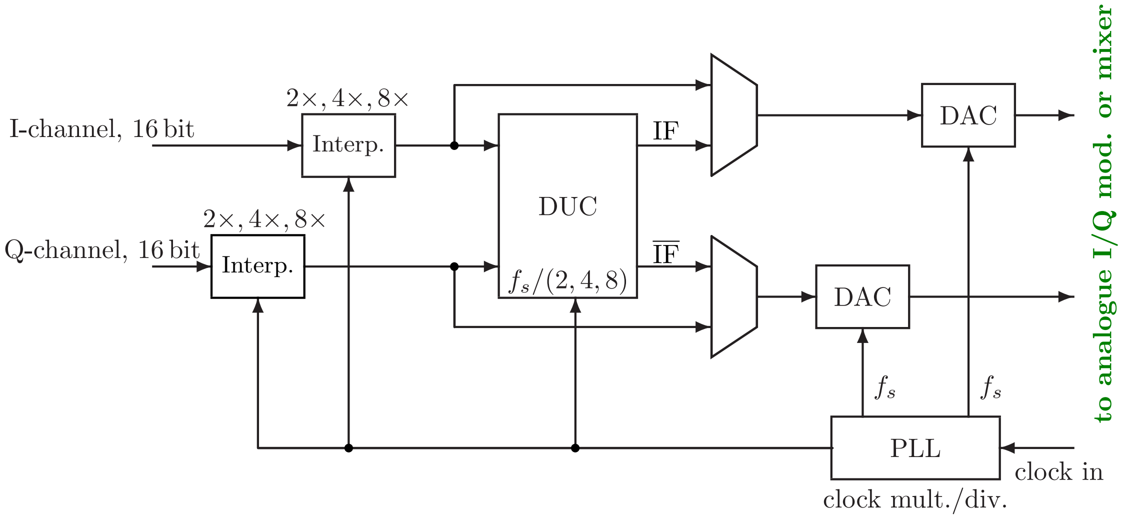


Figure 5.4: Functional diagram of DUC and DAC

Tab. 5.3 presents the key specifications of the DAC. Especially important are the linearity specifications, the high spurious-free dynamic range (SFDR) and the total harmonic distortion (THD).

ADC The analogue-to-digital converter (ADC) is required to possess high sampling-rate due to the large bandwidth of the amplifier output-signal, together with a high resolution. Further, high linearity requirements have to be fulfilled. The key performance figures of the used ADC AD6645 [80] are listed in Tab. 5.4.

Resolution	$f_{s,\max}$	min. SFDR	typ. SNR	typ. THD
16 bit	400 MHz	73 dBc	75 dB	-71 dB

Table 5.3: Key specifications of the DUC/DAC (AD9777)

Resolution	$f_{s,\max}$	min. SFDR	typ. SNR	max. input range
14 bit	105 MHz	89 dBc	75 dB	2,2 $\mathrm{V_{ss}}$

Table 5.4: Key specifications of the ADC (AD6645)

In order to fully exploit the dynamic range, the input signal must have the full voltage range of 2,2 $\mathrm{V_{ss}}$. This causes problems since the down-converter, if driven in a highly linear mode, is not capable of delivering such a high signal level. Therefore, the signal after the down-converter must be amplified which degrades the dynamic range due to the introduced noise. It is important to notice that the highly linear ADC and DAC do not determine the achieved dynamic range – this is determined by the up- and down-converters and the required signal-amplification after the down-converter.

Memory The sampled signal as well as the transmit-signal are saved in a fast memory. For the transmit-signal a bank of 1 M×32 bits of fast memory (up to 166 MHz) is available. Thus, if the in-phase and quadrature-phase (or, as in the implemented design, the magnitude and phase) are stored, requiring two bytes (14 bit resolution), a signal of 1 MSamples can be stored. This buffer can be read-out cyclically, therefore, periodic signals with a maximum period-length of 1 MSamples can be generated.

For storing the sampled signal a very large memory of up to 2 GB is available, allowing to save a data stream with a rate of 400 MB/s. Therefore, since 2 Bytes are required per input sample, a maximum sampling rate of the ADC of 100 MHz is allowed, which is at the same time also near the maximal sampling-rate of the AD6645 ADC.

5.2.1 Assembled Digital System

Fig. 5.5 illustrates the digital part of the implemented pre-distortion prototype in detail. In Appendix C.3 the hardware components are illustrated and briefly described.

A user-defined signal, generated in Matlab®, can be loaded via the DSP into the memory-section of the Sundance® SMT370 module [81]. In this prototype-design only the nonlinear AM-AM conversion is compensated, therefore, only the magnitude of the signal has to be pre-distorted. Thus, the memory is divided into two sections where the magnitude of the signal and the corresponding phase of the signal can be stored. The magnitude is then filtered by the pre-distortion filter, the phase is not modified. The parameters required by the pre-distortion filter, namely the model-parameters and the targeted linear gain, see (4.23) and (4.16), are computed using the sampled output signal of the RF-part, stored in the large memory of the Sundance® SMT351 module [82]. The pre-distorted signal magnitude and the (undistorted) phase are then used to calculate the in-phase (I-) and quadrature-phase (Q-) component. The pre-distortion filter and the calculation of the I/Q-components are implemented in the FPGA on the Sundance® SMT370 module and run in real-time with a clock-frequency of $f_T = 70\,\mathrm{MHz}$. Thus, a theoretical baseband bandwidth of 70 MHz is possible. The implemented pre-distortion filter design allows for a maximum clock-frequency of $f_T = 133\,\mathrm{MHz}$.

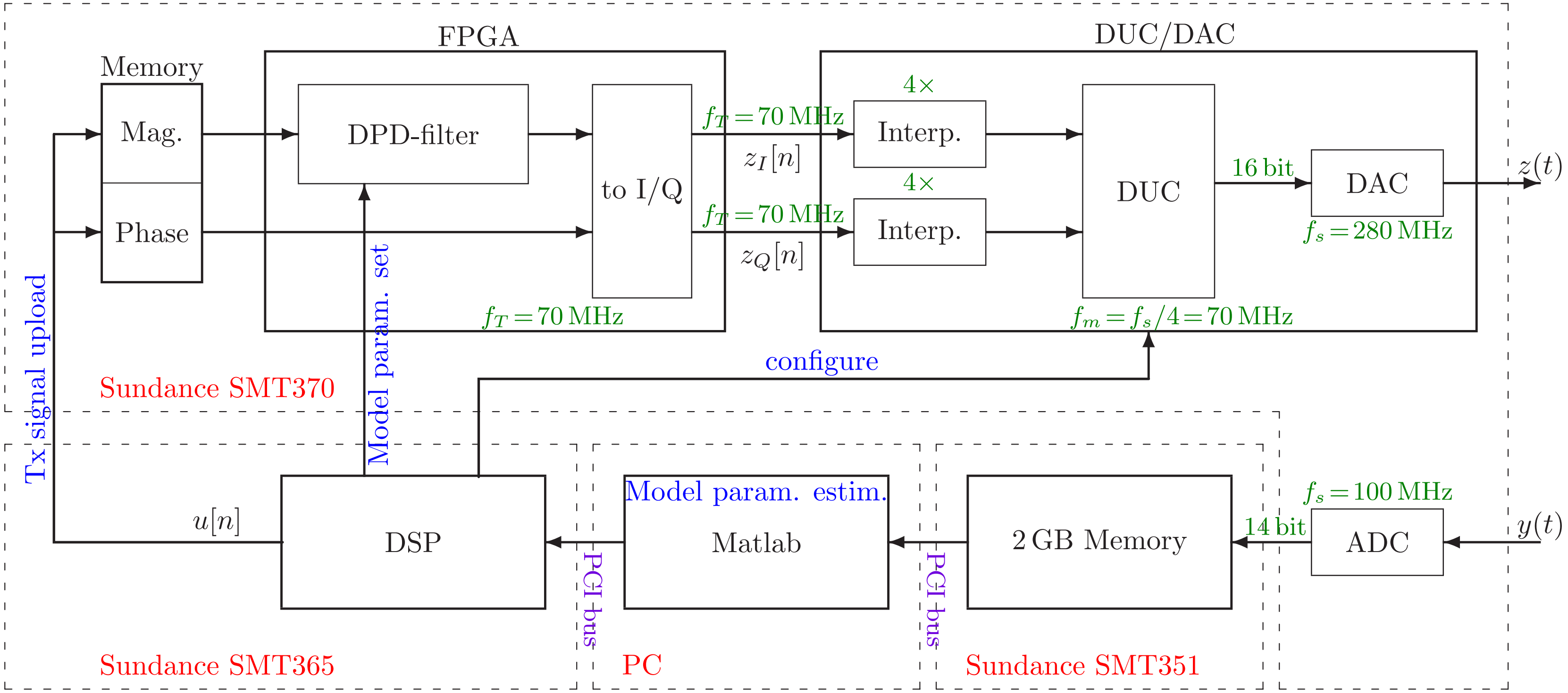


Figure 5.5: Digital part of the pre-distortion prototype in detail

After I/Q-mapping the I/Q-components are four-times interpolated and digitally modulated to a carrier at $f_m = 70\,\mathrm{MHz}$, required by the analogue RF-part. This digital signal is then converted to an analogue signal with a sampling rate of $f_s = 280\,\mathrm{MHz}$, corresponding to the data-rate after the interpolation filters. These functions are provided by the AD9777-DAC on the Sundance® SMT370 module. The parameters, the sampling-rate f_s, the interpolation-factors, as well as the modulation frequency f_m, are controlled by the DSP on the Sundance® SMT365 module [83]. This module communicates via a 20 MB/s control-bus with the SMT370 module and with the PC, concretely with Matlab® via the PCI [20] at the PC. The system can be fully controlled by the PC.

5.3 Radio-Frequency Part – Description

Fig. 5.6 illustrates the analogue radio-frequency (RF) part of the measurement-setup. It comprises an up-conversion stage, intended to translate an analogue signal, centred at 70 MHz, to the ISM-band at 2,45 GHz and a down-conversion stage, which translates an RF-signal, centred at 2,45 GHz to 70 MHz [63]. The important aspects for the pre-distortion setup are the required power-levels for a linear operation of the up- and down-converters, as well as the bandwidth-limitation to a 3 dB-bandwidth of $\Delta f_{3\,\mathrm{dB}} \approx 20\,\mathrm{MHz}$, which is caused by the bandpass filters (BPF) in the up- and down-converters. The power levels at the different positions in the setup are listed in Tab. 5.5. The distortion-free dynamic range of the up- and down-converter is larger than 50 dB with these input-power levels.

Location	P_1	P_2	P_3	P_4	P_5	P_6	P_7
approx. Power / [dBm]	-20	-17	0	$0, (-1)$	$29, (30)$	$-21, (-20)$	$-18, (-17)$

Table 5.5: Power levels at different positions, see Fig. 5.6

The Mini-Circuits power-amplifier ZJL-4HG [84] amplifies the output-signal of the up-converter to a sufficiently high power-level in order to drive the power-amplifier ZVE-8G [58] into saturation. This power amplifier is without input back-off (attenuation ATT=0 dB) at the 1 dB compression point ($P_5 = 30\,\mathrm{dBm}$), thus operating in the nonlinear region. After attenuation, the signal is down-converted to a centre-frequency of 70 MHz. Since the output-power of the down-conversion stage is very small ($P_7 = -18\,\mathrm{dBm} \div -17\,\mathrm{dBm}$), the signal must be amplified again in order to fully exploit the input-range of the ADC ($U_{\mathrm{ss,max.}} = 2,2\,\mathrm{V}$), cf. Tab. 5.4. This is achieved with two low-noise amplifiers (Avantek GPD-461/462/464 and Mini-Circuits ZHL-1042J [58]) and two attenuators, one of these being variable in order to adjust the output signal-level to the full input-range of the ADC of $2,2\,V_{\mathrm{ss}}$.

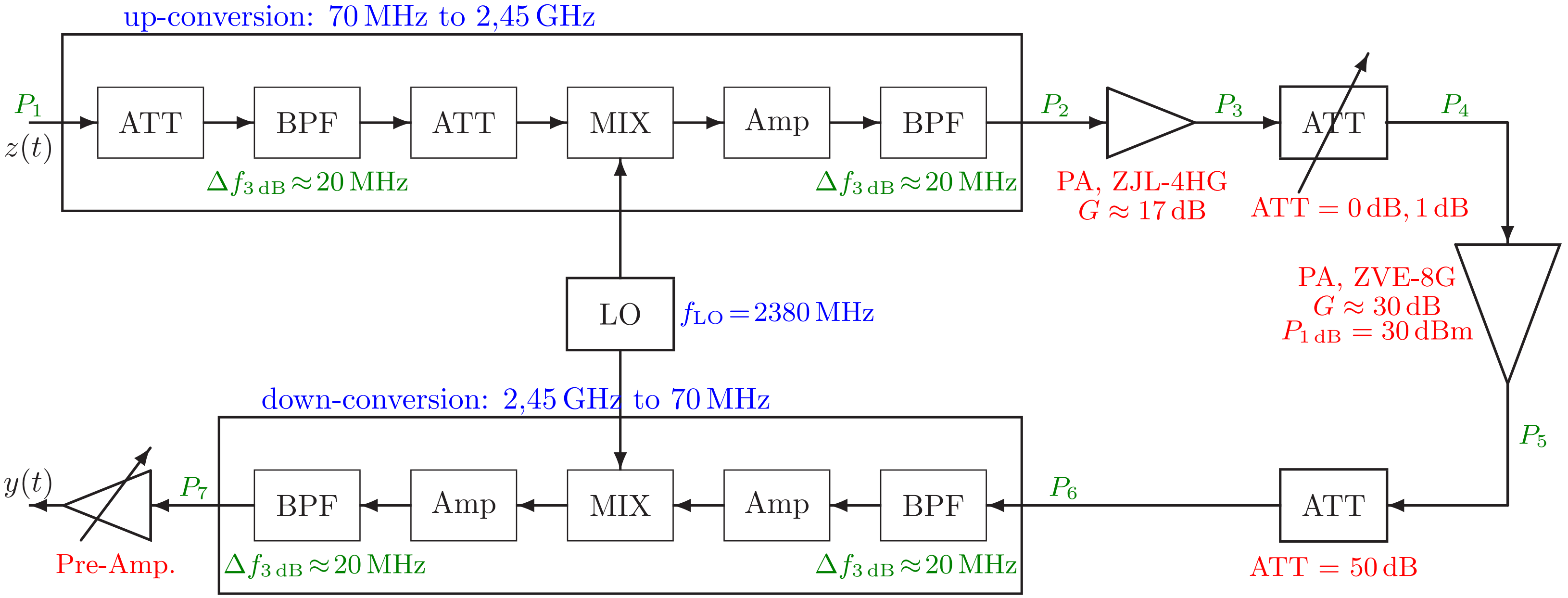

up-conversion: 70 MHz to 2,45 GHz
P_1
$z(t)$
ATT
BPF
ATT
MIX
Amp
BPF
$\Delta f_{3\,\mathrm{dB}} \approx 20\,\mathrm{MHz}$
$\Delta f_{3\,\mathrm{dB}} \approx 20\,\mathrm{MHz}$
P_2
PA, ZJL-4HG
$G \approx 17\,\mathrm{dB}$
P_3
ATT
ATT = 0 dB, 1 dB
P_4
PA, ZVE-8G
$G \approx 30\,\mathrm{dB}$
$P_{1\,\mathrm{dB}} = 30\,\mathrm{dBm}$
P_5
LO
$f_{\mathrm{LO}} = 2380\,\mathrm{MHz}$
down-conversion: 2,45 GHz to 70 MHz
$y(t)$
Pre-Amp.
P_7
BPF
Amp
MIX
Amp
BPF
$\Delta f_{3\,\mathrm{dB}} \approx 20\,\mathrm{MHz}$
$\Delta f_{3\,\mathrm{dB}} \approx 20\,\mathrm{MHz}$
P_6
ATT
ATT = 50 dB

Figure 5.6: Analogue RF-part of the pre-distortion prototype

5.4 Implementation Details

In this section, the implemented pre-distortion algorithm is described in more detail. The implementation is divided into two parts: the actual pre-distortion filter implementation and the model-parameter estimation. The pre-distortion filter is implemented in the FPGA and runs in real-time, whereas the model-parameter estimation is implemented in software on the PC. Such design supports the fact that the behaviour of the power amplifier is not changing very rapidly. It is sufficient to estimate the model parameters only after changing the operating conditions, e.g, using a different input power level.

This pre-distortion filter design is based on a very simple model for the power-amplifier. Only the nonlinear distortions of the amplitude of the signal are corrected. The design is intended to demonstrate only the implementation-feasibility of the pre-distortion algorithm, the complexity is intentionally kept minimal.

The power-amplifier is modelled as a memoryless system with only nonlinear AM-AM conversion. The model is a Taylor series with linear and third-order part,

$$y[n] = \mathbb{N}\big(u[n]\big) = \theta_1 u[n] + \theta_3 u[n]\big|u[n]\big|^2 = \Big(\theta_1\big|u[n]\big| + \theta_3\big|u[n]\big|^3\Big)e^{j\arg(u[n])} , \tag{5.1}$$

$u[n]$ being the input-signal and $y[n]$ the output-signal of the power-amplifier model. The phase of the input signal remains unchanged. Only two model-parameters have to be estimated. It is clear that only the third-order intermodulation products can be corrected with this model. The two parameters θ_1 and θ_3, along with the intended linear gain g, see (5.2), are determined by the modelling part.

5.4.1 Pre-distortion Filter Implementation

Fig. 5.7 illustrates the basic operation of the pre-distortion filter. Three iterations of the pre-distortion algorithm, based on the Secant method, see Section 4.3, are implemented in the FPGA. The first stage of the filter starts with two initial values, $z_0[n] = u[n]$ and $z_{-1}[n] = 0$. The operator $\mathbb{T}\big(z[n]\big)$ is, cf. (4.16),

$$\mathbb{T}\big(z[n]\big) = \mathbb{N}\big(z[n]\big) - g \cdot u[n] , \tag{5.2}$$

with the targeted linear gain g. The input $\mathbb{T}\big(z_{-1}[n]\big) = \mathbb{T}\big(0\big) = -g \cdot u[n]$, requires thus only one multiplication. The product $g \cdot u[n]$ has to be determined anyway for the following iterations. Therefore, the calculation of the starting value $\mathbb{T}\big(z_{-1}[n]\big)$ requires effectively only a sign-change of the product $g \cdot u[n]$.

Fig. 5.8 illustrates one filter-stage. Based on the four input signals $z_i[n]$, $z_{i-1}[n]$, $\mathbb{T}\big(z_{i-1}[n]\big)$ and $g \cdot u[n]$, the new value $z_{i+1}[n]$ is determined, an approximation of the ideal linearising signal $z[n]$, see (4.23). One stage requires the evaluation of the operator $\mathbb{T}(\cdot)$, which is basically determined by the selected model $\mathbb{N}(\cdot)$ for the power-amplifier. In the implemented design, the evaluation of this operator requires four real-valued multiplications (only the amplitude is taken into consideration), one addition, and one subtraction, cf. (5.1). Further, one additional multiplication, one division, and three subtractions have to be performed. The total amount of arithmetic operations of one stage is summarised in Tab. 5.6.

The amount of operations, especially the (costly) multiplications, are essentially determined by the power amplifier model. If higher-order terms, e.g., fifth- and seventh-order terms are also taken into account, along with memory-effects, the complexity grows quickly. If memory effects are considered additionally, the signal after the pre-distortion filter $z_I[n-1]$, $I = 3$ in

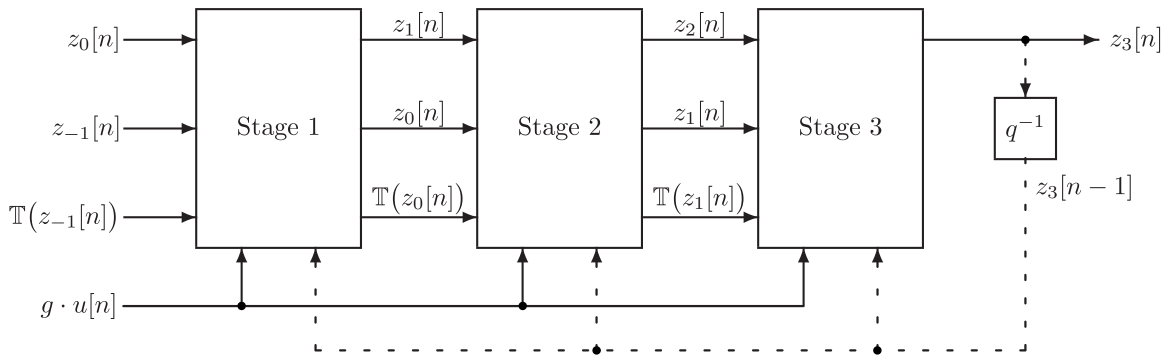


Figure 5.7: The implemented pre-distortion algorithm. The dashed feed-back is required if models with memory are used.

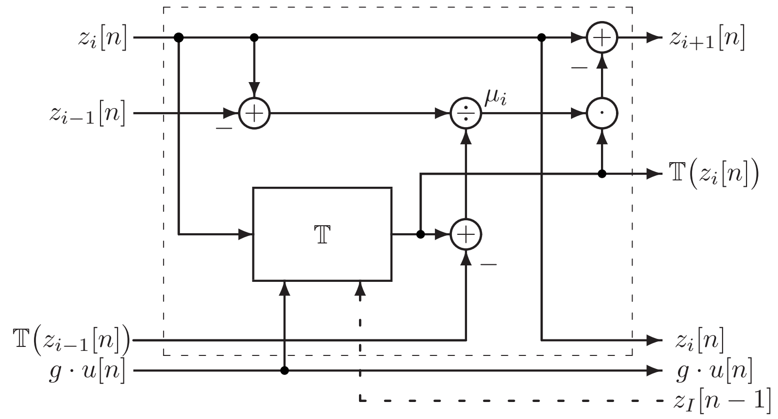


Figure 5.8: One stage of the implemented pre-distortion algorithm

Operations/stage	ADD,SUB	MUL	DIV
amount	5	5	1

Table 5.6: Amount of arithmetic operations per filter-stage

the presented design, must be fed back to each iteration stage, indicated by a dashed line in Fig. 5.7 and Fig. 5.8.

Division

The FPGA provides optimised hardware multipliers but does not provide optimised hardware dividers. The XILINX® LogiCore library provides an IP-core for a divider implementation [85] but it proves to be too costly in terms of resources. Therefore, an alternative method, based on the Newton-Raphson root-finding algorithm is used [20]. If a division

$$r = \frac{n}{d} = n \cdot \frac{1}{d} = n \cdot x \tag{5.3}$$

has to be performed, the task is to calculate $x = \frac{1}{d}$ and multiply the result with the numerator n. Rearranging terms gives

$$d - \frac{1}{x} = f(x) = 0 \tag{5.4}$$

which can be solved with the Newton-Raphson method:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} = x_i(2 - dx_i) \ , i \geq 0 \, , \ x_0 \, \text{given} \ . \tag{5.5}$$

The convergence rate of the Newton-Raphson algorithm is quadratic, therefore, it can be expected that few iterations are sufficient. Further, the starting value x_0 can be chosen freely and, thus, a list of optimised starting values can be produced. Based on the value of d, the optimal value x_0 can be chosen. If x_0 is further chosen to be a power of two, the multiplications with x_0 reduce to cheap shift-operations. In this way, the first iteration x_1 is computed without a multiplication.

The range of the possible values for a fractional number[1] [86] is divided into $N-1$ intervals $I_k \equiv [2^{-k} + \Delta; 2^{-(k-1)}], k = 1, 2, \ldots, N-1$, Δ being the resolution $\Delta = 2^{-(N-1)}$. The starting-value x_0 for each interval is then chosen to be $x_0 = 2^{k-1}$, thus, at the upper limit of the interval, the correct result is obtained with the starting-value. Tab. 5.7 shows a list of starting values, assuming that four bits are used to represent the involved signals. Here, the number d is given by a fractional 1.3 two's-complement representation and only positive values, ranging from 1 to Δ are taken into account. The resolution (or numerical value of the least significant bit) $\Delta = 2^{-3} = 1/8$.

k	I_k	Exact value, $x = \frac{1}{d}$	Starting value, $x_0 = 2^{k-1}$
1	$\left[\frac{5}{8}; 1\right]$	$\left[\frac{8}{5}; 1\right]$	$2^0 = 1$
2	$\left[\frac{3}{8}; \frac{4}{8}\right]$	$\left[\frac{8}{3}; 2\right]$	$2^1 = 2$
3	$\frac{2}{8}$	4	$2^2 = 4$
4	$\frac{1}{8}$	8	$2^3 = 8$

Table 5.7: Starting values for the Newton-Raphson method applied for performing a division $1/d$, d being represented by four bits and interpreted as a fractional number.

It can be easily shown that with these starting-values, the Newton-Raphson algorithm is guaranteed to converge, see Appendix C.1. Further, an error-analysis, cf. Appendix C.2, shows that after the second iteration, the relative error $\varepsilon_2 = \frac{x_2 - x}{x}$ is only 6, 25 %. The arithmetic cost for the division, if only two iterations are performed, is only two multiplications (the multiplications with the initial value in the first iteration are shift operations) and three

---

[1] A number x can be represented with N bits in I.Q-format, $I = 1, Q = N - 1$ as

$$x = -b_{N-1} + \sum_{k=1}^{N-1} b_{N-1-k} 2^{-k} \ , \ b_{N-1-k} \in \{0, 1\} \text{ and } -1 \leq x \leq 1 - 2^{-N+1} \ .$$

The resolution is $\Delta = 2^{-N+1}$, if $N = 4$, $\Delta = 2^{-3} = 0,125$.

subtractions. With the multiplication of the numerator, three multiplications in total are necessary.

5.4.2 Model-parameter Estimation

Fig. 5.9 shows how the model-parameter estimation, based on the stored output signal $y[n]$ and the known input signal $u[n]$, is performed. Since the ADC-converter samples a signal centred at $f_m = 70\,\text{MHz}$ and bandwidth $\Delta f \approx 20\,\text{MHz}$ with a sampling-frequency of $f_s = 100\,\text{MHz}$, the digital signal results centred at the frequency $f_m = 30\,\text{MHz}$, corresponding to a normalised frequency $\theta_m = \frac{f_m}{f_s} = 0,3$ (this is in fact an aliased spectrum). After bandpass-filtering with a digital bandpass-filter (BPF) and demodulation, the baseband signal is obtained. This output-baseband signal is then synchronised (aligned in the time-domain) with the input-signal (via a correlation the experienced delay is estimated). After an interpolation, which reduces the sampling-rate by a factor of four and at the same time the noise (implicit resolution enhancement), the model parameters (θ_1, θ_3) for the simple model (5.1) are estimated using the least-squares method. These operations are performed on the PC, the estimated parameters then transferred via the DSP to the FPGA, where the pre-distortion filter is implemented.

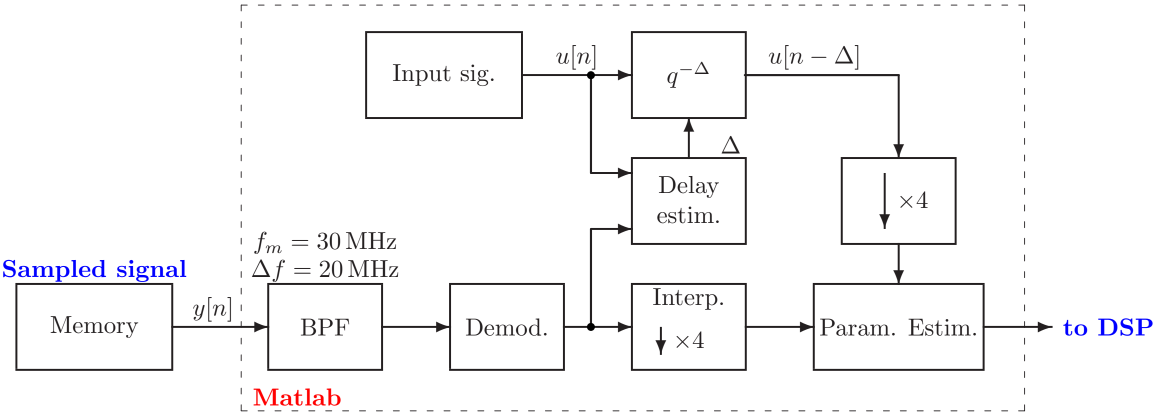


Figure 5.9: Model-parameter estimation

The model-parameters can be transferred to the FPGA without the necessity to interrupt the pre-distortion filter, running continuously on the FPGA. In this way, the model-parameter estimation can be performed in the background, either on the PC or directly on the DSP, once a new output-data record is available in the memory.

5.5 Measurements and Performance Results

In this section, measurement results illustrating the performance of the pre-distortion filter implementation, running in real-time on the FPGA, as well as a short summary, reporting the invested FPGA-resources, are presented.

5.5.1 Measurement Results: Floating-point vs. Fixed-point

The pre-distortion method based on the secant-algorithm is capable to linearise a nonlinear dynamic system. Simulations in Section 4.3 and especially measurements on a microwave high-power amplifier, reported in Section 4.4.3, proved the functionality. These results are obtained by double-precision calculations performed on a PC using Matlab® , i.e., the signal after the pre-distortion filter is calculated in advance, based on the preceding modelling of the power amplifier, and then used to excite the power amplifier.

In an FPGA implementation the pre-distortion filter is implemented using a very restricted number representation, i.e., instead of floating-point numbers with double-precision, fixed-point numbers are used. In the implemented design, a fixed-point number representation in fractional 1.17-format is used. Hence, 18 bits are used to represent a signal. This specific choice is motivated by the hardware-multipliers on the FPGA which can handle 18 bit signed values.

Before programming the FPGA, a comparison between the floating-point implementation of the pre-distortion filter (in Matlab®) and the VHDL[2]-implementation is performed. The VHDL-implementation of the pre-distortion filter is in fact an alternative description of the algorithm. In contrast to the double-precision Matlab® implementation the signals are represented and the arithmetic operations are performed using a fixed-point representation. Here, one or more periods of the signal at the output of the pre-distortion filter are pre-computed (off-line) with Matlab® and with ModelSim®[3]

Fig. 5.10 illustrates the performance of the pre-distortion algorithm based on the simple third-order model (5.1) describing only the nonlinear AM-AM conversion. The signal is a multitone signal with 5 MHz bandwidth. After the estimation of the model-parameters (θ_1, θ_2) the signal after the pre-distortion filter is calculated using Matlab® employing three iterations with the Secant method. This signal is then used to excite the RF-part of the measurement setup. A spectrum analyser measures the output-signal of the pre-amplifier after the down-converter, see Fig. 5.6.

Fig. 5.10 is a measurement result of the spectrum analyser and shows the excellent performance of the pre-distortion algorithm using floating-point number representation. The third-order intermodulation products are compensated perfectly. Approximately 15 dB of interference suppression are achieved.

Fig. 5.11 shows a comparison of the VHDL-description (fixed-point accuracy, calculation is done in ModelSim®) and the Matlab® description (double-precision) of the pre-distortion filter. Basically no performance degradation can be observed, proving the robustness of the derived pre-distortion method.

5.5.2 Measurement Results: Real-Time Implementation

Fig. 5.12 shows the measured output spectra, now with the pre-distortion filter running in real-time on the FPGA. In this setup, the pre-amplifier in front of the AD-converter is not used, therefore the power level decreases approximately by 20 dB, but with the advantage that the dynamic range is significantly increased and reaches approx. 57 dB. The disadvantage is that the input signal of the AD-converter is significantly smaller as the allowed range of 2,2 V_{ss}, resulting in a poorer SNR at the output of the AD-converter due to the introduced quantisation noise. In this case the loss due to the introduced noise by the pre-amplifier is

---

[2] VHDL or Very high speed integrated circuit Hardware Description Language, is commonly used as a design-entry language for FPGAs and ASICs in electronic design automation of digital circuits. VHDL was originally developed on demand of the US Department of Defense in order to document the behaviour of the ASICs that supplier companies were including in equipment, i.e., VHDL was developed as an alternative to huge, complex manuals which were subject to implementation-specific details.

[3] ModelSim® is a bit-true simulation and debug environment from Mentor Graphics®[87] for VHDL designs.

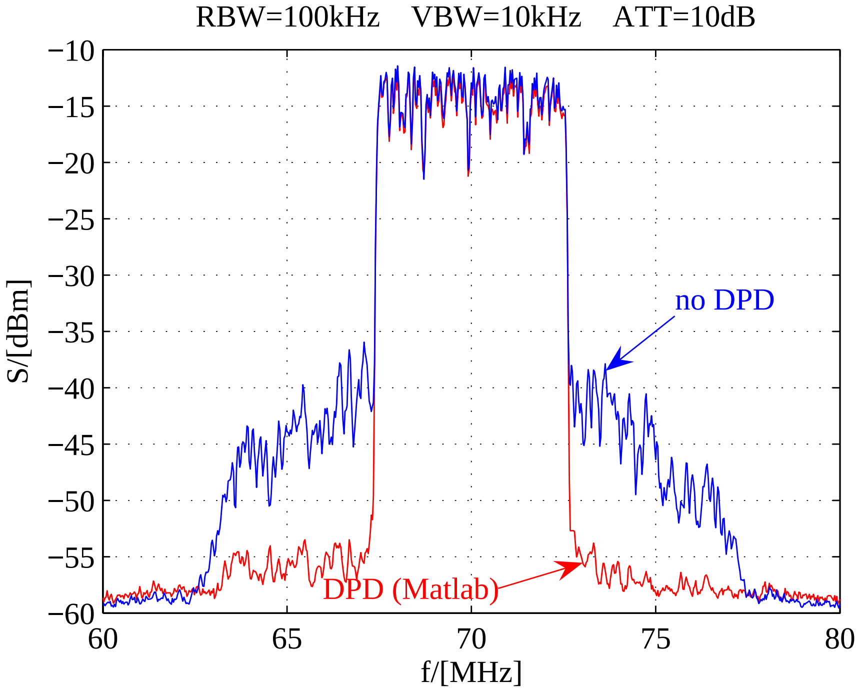


Figure 5.10: Measured output-spectra at 70 MHz intermediate frequency (IF), with and without pre-distortion: Signal after the pre-distortion filter is calculated in double-precision floating-point arithmetic.

larger as the loss due to the quantisation noise.

The model parameters are estimated once the power amplifier has reached a thermal steady-state. Then the model parameters are delivered to the pre-distortion filter and the pre-distortion filter, implemented in the FPGA, is activated. If the input power-level remains constant, no new model-parameter estimation is necessary, the performance does not degrade significantly over time (up to several hours).

A significant suppression of the third-order intermodulation products of up to 15 dB is achieved, compared with an input power back-off (IBO) of 1 dB which assures the same power level with and without pre-distortion. Essentially no difference between the real-time operation, working with fixed-point accuracy, and the Matlab® calculation, where the signal after the pre-distortion filter is pre-determined with double precision, exists. This proves that the implemented pre-distortion algorithm is robust against rounding errors. Fig. 5.13 illustrates the spectral regrowth and the compensation in more detail.

In order to achieve the same low out-of-band distortions applying only input power back-off, the input power has to be reduced significantly. As can be seen in Fig. 5.14, a back-off of approximately 9 dB is necessary to guarantee equal out-of-band distortions.

The measurement results reported here prove not only the principal functionality of the developed pre-distortion algorithm, but show also that the algorithm is robust enough to deliver excellent performance results in a fixed-point implementation.

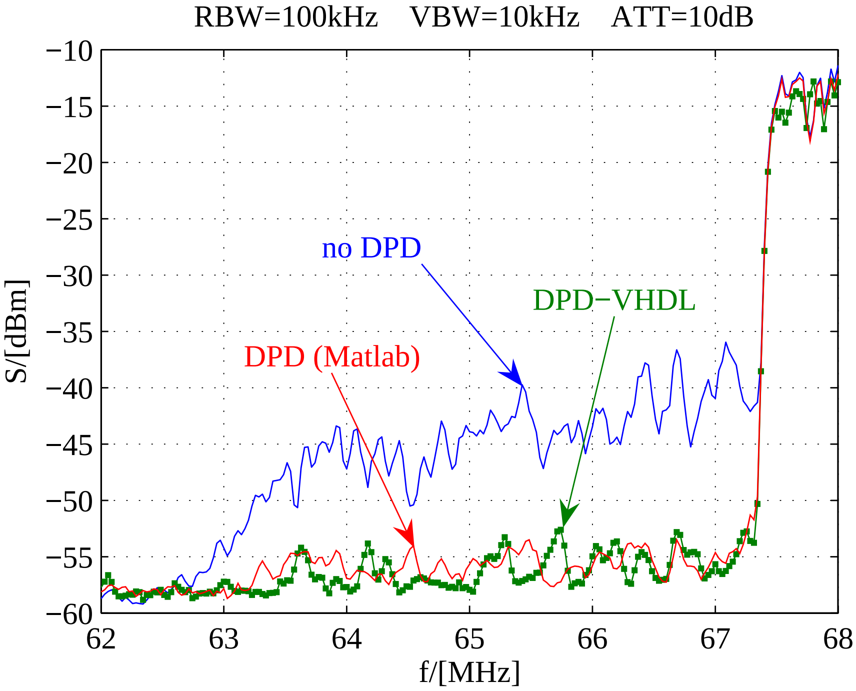


Figure 5.11: Measured output-spectra at 70 MHz IF, with and without pre-distortion: Signal after the pre-distortion filter is calculated in Matlab® and via a simulation of the VHDL-description using fixed-point arithmetic.

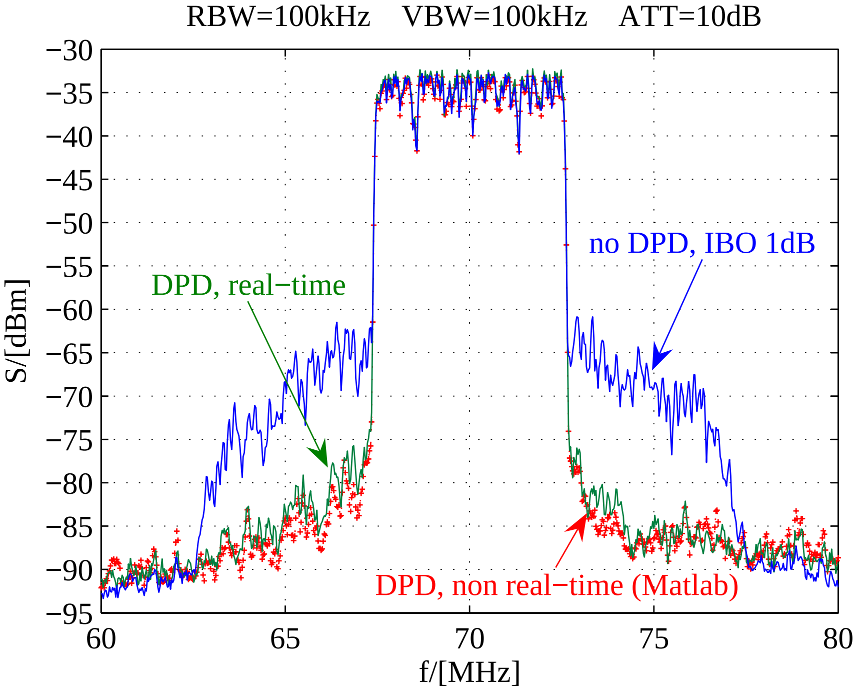


Figure 5.12: Measured output-spectra at 70 MHz IF: Comparison of input-back off (IBO) and digital pre-distortion

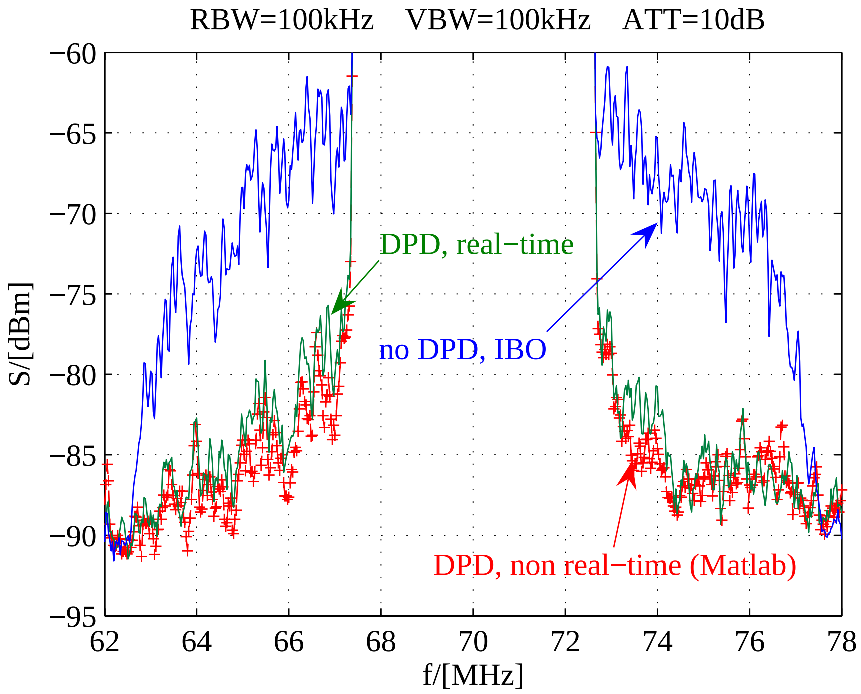


Figure 5.13: Measured output-spectra at 70 MHz IF: Comparison of input-back off (IBO) and digital pre-distortion, detailed view

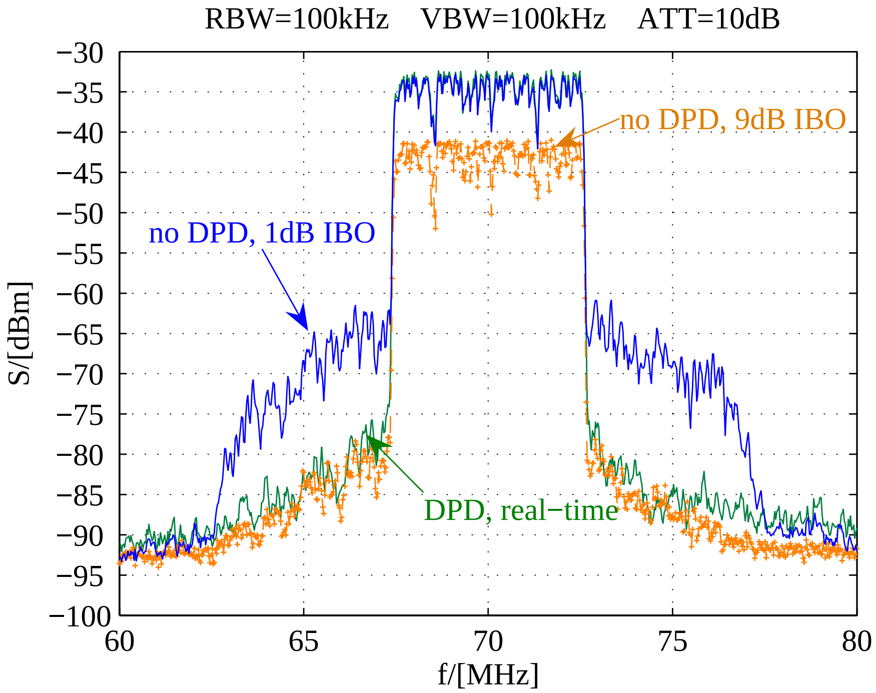


Figure 5.14: Measured output-spectra at 70 MHz IF: Comparison of input-back off (IBO) and digital pre-distortion for achieving equal out-of-band distortions

5.5.3 Used FPGA Resources

Finally, the used FPGA-resources of the implemented pre-distortion filter are reported in Tab. 5.8. The design can be clocked with a maximum frequency of $f_{T,\max} = 133\,\mathrm{MHz}$. Only approx. 50 % of the available resources are used. Here, an FPGA slice is a subdivision of a CLB.

resource	Multipliers	Flip-flop slices	Slices	Look-up tables
# used	25	4.517	2.918	1.241
usage in %	62	44	56	12

Table 5.8: Used FPGA resources, absolute numbers and used percentage of the total available resources

Since not all FPGA resources are consumed by the design, further enhancements, e.g., using models with higher-order nonlinear terms and/or memory effects, are possible.

5.6 Discussion

The development of a prototype-implementation of the pre-distortion algorithm, introduced in Chapter 4, was documented in this chapter. The signal processing part is divided in two parts, the pre-distortion filter which runs in real-time on an FPGA and the model-parameter estimation, which is performed on the PC. A minimal design, based on a very simple power amplifier model, describing only nonlinear AM-AM conversion, has been implemented.

The aims of this design were to show that the presented pre-distortion algorithm is

- not too complex for an implementation in hardware, and
- robust enough to perform also well in a fixed-point implementation.

Both goals have been reached, as measurement results, presented in Section 5.5, prove.

The design uses approx. 50 % of the FPGA-resources. This, together with a modular implementation, enables eventual future extensions of the prototype.

Chapter 6

Conclusions

This thesis addressed digital pre-distortion for linearisation of nonlinear and dynamic microwave power amplifiers. The aim was the development of a linearisation algorithm which is capable of linearising a variety of power amplifiers. Such algorithm had to be implemented in a prototype system in order to prove its practical value.

The devised linearisation method can be considered as very flexible, capable of being applied in a plurality of applications with different degrees of complexity. These applications can range from power amplifier linearisation in mobile communication base-stations of the second generation to multichannel arrangements in base-stations of the third- and future generations, as well as applications in high data-rate wireless local area networks.

En route to the implementation of a prototype pre-distortion system, four main tasks have been addressed:

1. Modelling of the microwave power amplifier
2. Low complex and robust identification of the model parameters
3. Development of a pre-distortion algorithm
4. Implementation in signal processing hardware

Various models for nonlinear and dynamic microwave power amplifies have been evaluated. The aim was to identify possible accurate and low-complex candidates for power amplifier models. Based on time-domain measurements on different microwave power amplifiers it was shown that low-order nonlinearities, i.e., up to nonlinear order seven, are sufficient for an accurate description of the nonlinear effects. The investigated power amplifiers showed further that memory effects are rather concentrated, meaning that the kernel-lengths of, e.g., a Volterra model, are short. It was further shown that low complex models, such as Wiener- or Hammerstein models, are adequate.

Adaptive gradient-type identification schemes for the identification of the parameters of a Volterra- and Wiener systems have been presented. The stability analysis of the derived adaptive algorithms was performed in a deterministic context. The common stochastic analysis fails because important information regarding the statistics of the signals and required partial knowledge of system characteristics is in practical applications missing. The deterministic analysis provided furthermore very strong results about the stability of the devised algorithms which cannot be provided by a conventional stochastic analysis.

To develop an analytical solution for the pre-equaliser is rarely possible in practice. Therefore, an iterative procedure for the determination of the output signal of the pre-distortion filter has been devised. It is based on the Secant method for root-finding and applicable for all investigated power amplifier models, providing an approximation to the exact solution for the pre-equaliser. The presented method is therefore a powerful tool for solving the equalisation problem in an approximate way. The derived method proved further to be advantageous in terms of accuracy and complexity if compared with a known method based on a fixed-point formulation of the pre-distortion problem.

A pre-distortion prototype system, based on the devised linearisation algorithm, has been developed. The core of this hardware system is a real-time implementation of the linearisation algorithm in an FPGA. With this experimental setup it has been shown that the derived pre-distortion method admits also a practical implementation where only a limited amount of resources and limited arithmetic accuracy is available. Measurement results with microwave power amplifiers proved the excellent performance of the implemented algorithm.

Appendix A

Appendix: Adaptive Identification

Various definitions and derivations for Chapter 3 are presented.

A.1 Derivation of (3.40)

The local energy relation reads

$$\|\tilde{\boldsymbol{\theta}}_n\|_2^2 + \left|\bar{e}_a[n]\right|^2 = \|\tilde{\boldsymbol{\theta}}_{n-1}\|_2^2 + \left|\bar{\bar{v}}[n]\right|^2 , \tag{A.1}$$

using the definitions

$$\begin{aligned} \bar{e}_a[n] &= \sqrt{\bar{\mu}[n]}e_a[n] && \text{(A.2)} \\ \bar{\bar{v}}[n] &= \sqrt{\bar{\mu}[n]}\bar{v}[n] . && \text{(A.3)} \end{aligned}$$

The finite time-horizon $n = 0, \ldots, N$ is considered. For a derivation of the global passivity relation using the feedback structure, the following vectors-signals are defined:

$$\begin{aligned} \bar{\mathbf{e}}_{a,N} &= \left[\sqrt{\bar{\mu}[0]}e_a[0], \sqrt{\bar{\mu}[1]}e_a[1], \ldots, \sqrt{\bar{\mu}[N]}e_a[N]\right]^T && \text{(A.4)} \\ \bar{\bar{\mathbf{v}}}_N &= \left[\sqrt{\bar{\mu}[0]}\bar{v}[0], \sqrt{\bar{\mu}[1]}\bar{v}[1], \ldots, \sqrt{\bar{\mu}[N]}\bar{v}[N]\right]^T . && \text{(A.5)} \end{aligned}$$

Summation over the time-horizon $n = 0, \ldots, N$ of (A.1) gives

$$\|\tilde{\boldsymbol{\theta}}_N\|_2^2 + \|\bar{\mathbf{e}}_{a,N}\|_2^2 = \|\tilde{\boldsymbol{\theta}}_{-1}\|_2^2 + \|\bar{\bar{\mathbf{v}}}_N\|_2^2 . \tag{A.6}$$

A lossless map $\bar{\mathbb{T}}$, with $\|\bar{\mathbb{T}}\|_2^2 = 1$, transforms the input-vectors $\tilde{\boldsymbol{\theta}}_{-1}$ and $\bar{\bar{\mathbf{v}}}_N$ to the output-vectors $\tilde{\boldsymbol{\theta}}_N$ and $\bar{\mathbf{e}}_{a,N}$.

With the definitions

$$\begin{aligned} \bar{\mathbf{v}}_N &= \left[\sqrt{\bar{\mu}[0]}v[0], \sqrt{\bar{\mu}[1]}v[1], \ldots, \sqrt{\bar{\mu}[N]}v[N]\right]^T && \text{(A.7)} \\ \mathbf{G}_N &= \operatorname{diag}\left\{\left(1 - \frac{\mu[0]}{\bar{\mu}[0]}\right), \ldots, \left(1 - \frac{\mu[N]}{\bar{\mu}[N]}\right)\right\} && \text{(A.8)} \\ \mathbf{A}_N &= \operatorname{diag}\left\{\frac{\mu[0]}{\bar{\mu}[0]}, \ldots, \frac{\mu[N]}{\bar{\mu}[N]}\right\} , && \text{(A.9)} \end{aligned}$$

the system equations read

$$\begin{aligned} \begin{bmatrix} \tilde{\boldsymbol{\theta}}_N \\ \bar{\mathbf{e}}_{a,N} \end{bmatrix} &= \bar{\mathbb{T}} \begin{bmatrix} \tilde{\boldsymbol{\theta}}_{-1} \\ \bar{\mathbf{v}}_N \end{bmatrix} , \text{ with } \|\bar{\mathbb{T}}\|_2^2 = 1 \qquad (A.10) \\ \bar{\bar{\mathbf{v}}}_N &= \mathbf{A}_N \bar{\mathbf{v}}_N - \mathbf{G}_N \bar{\mathbf{e}}_{a,N} \; . \qquad (A.11) \end{aligned}$$

The gain of the feedback path is

$$g_{\text{FB}} = \|\mathbf{G}_N\|_2^2 = \max\big|\lambda(\mathbf{G}_N)\big|^2 = \max_{n=0,\dots,N} \left|1 - \frac{\mu[n]}{\bar{\mu}[n]}\right|^2 , \qquad (A.12)$$

where $\lambda(\mathbf{G}_N)$ is the spectrum of the matrix $\mathbf{G}_N$. Using the small gain theorem [69], the system is guaranteed to be stable if

$$g_{\text{FB}}\, g_{\text{FF}} < 1 , \qquad (A.13)$$

and, since $g_{\text{FF}} = \|\bar{\mathbb{T}}\|_2^2 = 1$, global stability is assured if

$$\max_{n=0,\dots,N} \left|1 - \frac{\mu[n]}{\bar{\mu}[n]}\right| < 1 \; . \qquad (A.14)$$

Thus, if

$$0 < \mu[n] < 2\bar{\mu}[n] \; , \; n = 0, \dots, N \; , \qquad (A.15)$$

the gradient algorithm is globally stable [13, 14].

A.2 Derivation of (3.67) and (3.68)

Taking the norm on both sides of the update-equation (the derivation of the passivity relation for the update-equation (3.64) is similar)

$$\tilde{\mathbf{g}}_n^{'} = \tilde{\mathbf{g}}_{n-1}^{'} - \mu_g[n]\big(e_{a,g}[n] + v_g[n]\big)\mathbf{x}_n^{'H} \qquad (A.16)$$

and simple manipulation results in

$$\begin{aligned} &\|\tilde{\mathbf{g}}_n^{'}\|_2^2 - \|\tilde{\mathbf{g}}_{n-1}^{'}\|_2^2 + \mu_g[n]\big|e_{a,g}[n]\big|^2 - \mu_g[n]\big|v_g[n]\big|^2 = \\ &\mu_g[n]\big|e_{a,g}[n] - v_g[n]\big|^2 \Big(\mu_g[n]\|\mathbf{x}_n^{'}\|_2^2 - 1\Big) \end{aligned} \qquad (A.17)$$

which gives, if

$$\Big(\mu_g[n]\|\mathbf{x}_n^{'}\|_2^2 - 1\Big) < 0 \; , \qquad (A.18)$$

(cf. (3.65)) the passivity relation (3.67).

A.3 Derivation of (3.73)

Multiplication of the update-equation

$$\tilde{\mathbf{h}}_n = \tilde{\mathbf{h}}_{n-1} - \mathbf{M}_n\big(e_a[n] + v_e[n]\big)\boldsymbol{\varphi}_n^H \; , \qquad (A.19)$$

with $\mathbf{M}_n^{-1/2}$ yields

$$\mathbf{M}_n^{-1/2}\tilde{\mathbf{h}}_n = \mathbf{M}_n^{-1/2}\tilde{\mathbf{h}}_{n-1} - \mathbf{M}_n^{1/2}\big(e_a[n] + v_e[n]\big)\boldsymbol{\varphi}_n^H \; . \qquad (A.20)$$

Taking the squared ℓ_2-norm on both sides results in

$$\begin{aligned}\tilde{\mathbf{h}}_n^H \mathbf{M}_n^{-1} \tilde{\mathbf{h}}_n = \tilde{\mathbf{h}}_{n-1}^H \mathbf{M}_n^{-1} \tilde{\mathbf{h}}_{n-1} &+ \boldsymbol{\varphi}_n \mathbf{M}_n \boldsymbol{\varphi}_n^H \left|e_a[n] + v_e[n]\right|^2 \\ &- 2\left|e_a[n]\right|^2 - e_a[n] v_e^*[n] - e_a^*[n] v_e[n] \; .\end{aligned} \tag{A.21}$$

If

$$0 < \boldsymbol{\varphi}_n \mathbf{M}_n \boldsymbol{\varphi}_n^H < 1 \tag{A.22}$$

$$\tilde{\mathbf{h}}_n^H \mathbf{M}_n^{-1} \tilde{\mathbf{h}}_n \le \tilde{\mathbf{h}}_{n-1}^H \mathbf{M}_n^{-1} \tilde{\mathbf{h}}_{n-1} - \left|e_a[n]\right|^2 + \left|v_e[n]\right|^2 \tag{A.23}$$

results. This gives the local passivity relation (3.73).

A.4 Derivation of (3.86)

Insertion of the modified noise

$$\bar{v}_e[n] = \frac{\alpha[n]}{\bar{\alpha}[n]} v_e[n] - \left(1 - \frac{\alpha[n]}{\bar{\alpha}[n]}\right) e_a[n] \tag{A.24}$$

in

$$\sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|\bar{v}_e[n]\right|^2} = \sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|\frac{\alpha[n]}{\bar{\alpha}[n]} v_e[n] - \left(1 - \frac{\alpha[n]}{\bar{\alpha}[n]}\right) e_a[n]\right|^2} \tag{A.25}$$

results in the upper bound

$$\sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|\bar{v}_e[n]\right|^2} \le \sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|\frac{\alpha[n]}{\bar{\alpha}[n]} v_e[n]\right|^2} + \sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|\left(1 - \frac{\alpha[n]}{\bar{\alpha}[n]}\right) e_a[n]\right|^2} \; . \tag{A.26}$$

This can be upper bounded further

$$\begin{aligned}&\sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|\frac{\alpha[n]}{\bar{\alpha}[n]} v_e[n]\right|^2} + \sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|\left(1 - \frac{\alpha[n]}{\bar{\alpha}[n]}\right) e_a[n]\right|^2} \\ &\le \delta_N \sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|v_e[n]\right|^2} + \gamma_N \sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|e_a[n]\right|^2} \; ,\end{aligned} \tag{A.27}$$

whereby

$$\delta_N = \max_{n=0,\ldots,N} \frac{\alpha[n]}{\bar{\alpha}[n]} \tag{A.28}$$

$$\gamma_N = \max_{n=0,\ldots,N} \left|1 - \frac{\alpha[n]}{\bar{\alpha}[n]}\right| \; . \tag{A.29}$$

Insertion of the upper bound (A.27) in

$$\sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|e_a[n]\right|^2} \le \sqrt{\tilde{\mathbf{h}}_{-1}^H \mathbf{M}^{-1} \tilde{\mathbf{h}}_{-1}} + \sqrt{\sum_{n=0}^{N} \bar{\alpha}[n] \left|\bar{v}_e[n]\right|^2} \; . \tag{A.30}$$

yields after simple manipulations the global passivity relation (3.86).

Appendix B

Appendix: Linearisation by Pre-distortion

B.1 The Contraction Mapping Theorem

The contraction mapping theorem can be stated as follows [49, 48]:

Contraction Mapping Theorem. *If* $\mathbb{T}$ *is a contraction mapping on a closed subset* $\mathscr{S}$ *of a Banach space*[1]*, there is a unique vector* $z \in \mathscr{S}$ *satisfying* $z = \mathbb{T}(z)$*. Furthermore,* z *can be obtained with the method of successive approximation starting from an arbitrary initial vector in* $\mathscr{S}$*.*

The definition of a contraction mapping is

Definition. *Let* $\mathscr{S}$ *be a subset of a normed space* $\mathcal{S}$ *and let* $\mathbb{T}$ *be a transformation mapping* $\mathscr{S}$ *into* $\mathscr{S}$*. Then* $\mathbb{T}$ *is said to be a contraction mapping if there is an* $\alpha, 0 \leq \alpha < 1$ *such that* $\|\mathbb{T}(z_1) - \mathbb{T}(z_2)\| \leq \alpha \|z_1 - z_2\|$ *for all* $z_1, z_2 \in \mathscr{S}$*.*

---

[1] A Banach space is a complete normed linear vector space. A normed linear vector space $\mathcal{X}$ is said to be complete if every Cauchy sequence from $\mathcal{X}$ has a limit in $\mathcal{X}$.

Appendix C

Appendix: Prototype Implementation

C.1 Convergence of the Newton-Raphson Method Applied for Division

A method to prove the convergence for the Newton-Raphson algorithm is based on the Contraction Mapping Theorem, see Appendix B.1. Starting point is the iteration-rule (5.5),

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} = x_i(2 - d \cdot x_i) \;, i \geq 0 \,,\; x_0 \,\text{given} \;. \tag{C.1}$$

This can be viewed as a method of successive approximation applied to the function

$$g(x) \equiv x - \frac{f(x)}{f'(x)} = x(2 - d \cdot x). \tag{C.2}$$

A fixed-point of $g(x)$ is thus a solution to $f(x) = 0$. Differentiating with respect to x gives

$$g_d^{'}(x) = 2(1 - d \cdot x). \tag{C.3}$$

The index d denotes that $g^{'}(x)$ is parameterised by d. For convergence,

$$\left|g_d^{'}(x)\right| < 1 \;, \tag{C.4}$$

resulting in the limits for x

$$\frac{1}{2d} < x < \frac{3}{2d} \;. \tag{C.5}$$

If x is in this interval, $g(x)$ is contractive. If, e.g., $d \in \left[\frac{3}{8}; \frac{1}{2}\right]$, corresponding to the second row in Tab. 5.7, the starting value $x_0 = 2$ is in the corresponding intervals $I_1 : \frac{4}{3} < x < 4$ for $d = \frac{3}{8}$ and $I_2 : 1 < x < 3$ for $d = \frac{1}{2}$, thus being in the contractive region of $g(x)$. It has further to be shown that the function $g(x)$ maps the intervals also in regions where $g(x)$ is contractive. As can easily be shown,

$$I_1 \xrightarrow{g(\cdot), d=3/8} \left[2; \frac{8}{3}\right] \in I_1 \tag{C.6}$$

$$I_2 \xrightarrow{g(\cdot), d=1/2} \left[\frac{3}{2}; 2\right] \in I_2 \;, \tag{C.7}$$

$g(x)$ remains therefore contractive. This can be done also for the first row of Tab. 5.7, thus, the method converges for all four possibilities.

C.2 Error-Analysis of the Newton-Raphson Method Applied for Division

The relative error ε_i at each iteration, assuming that $x = \frac{1}{d}$ is the correct value, is easily computed as

$$\varepsilon_i = \frac{x_i - x}{x} = d \cdot x_i - 1 \; . \tag{C.8}$$

Corresponding to Tab. 5.7, the worst-case values of d are the lower limits in each interval, i.e., $d_{\mathrm{wc}} = 2^{-k} + \Delta$ with $k \in \{1, N-1\}$ if N bits are used for the number representation, and $\Delta = 2^{-(N-1)}$. The correct value would be $x = \frac{1}{d_{\mathrm{wc}}} = \frac{1}{2^{-k}+\Delta} \approx \frac{1}{2^{-k}} = 2^k$, assuming that Δ is small (large N) and $k \ll N$. The initial value is $x_0 = 2^{k-1}$, thus, if $k \ll N$ and $N \gg$, the largest errors result. With only the initial value, the relative error is

$$\varepsilon_0 = d_{\mathrm{wc}} \cdot x_0 - 1 = \left(2^{-k} + \Delta\right)2^{k-1} - 1 \approx 2^{-1} - 1 = -50\,\% \; . \tag{C.9}$$

The first iteration results in

$$x_1 = x_0(2 - d_{\mathrm{wc}} \cdot x_0) = 2^{k-1}\left(2 - (2^{-k} + \Delta)2^{k-1}\right) \approx 2^k\left(1 - 2^{-2}\right) \; , \tag{C.10}$$

the relative error being thus

$$\varepsilon_1 = \frac{x_1 - x}{x} = -2^{-2} = -\frac{1}{4} = -25\,\% \; . \tag{C.11}$$

The second iterations reduces the error to

$$\varepsilon_2 = \frac{x_2 - x}{x} = -2^{-4} = -\frac{1}{16} = -6,25\,\% \; , \tag{C.12}$$

according to the quadratic convergence rate of the Newton-Raphson method.

C.3 The Prototyping-Hardware

Fig. C.1 shows a picture of the Sundance® SMT310Q carrier-board [88]. It is used to carry the processing modules (SMT365, SMT370-AC, SMT351-G) in the four available TIM-40 slots[1]. It provides the interface with the host PC via the PCI (Peripheral Component Interface). The V363EPC is a bridge-chip responsible for connecting the host PCI-bus to various devices on the PCI-card bus-system.

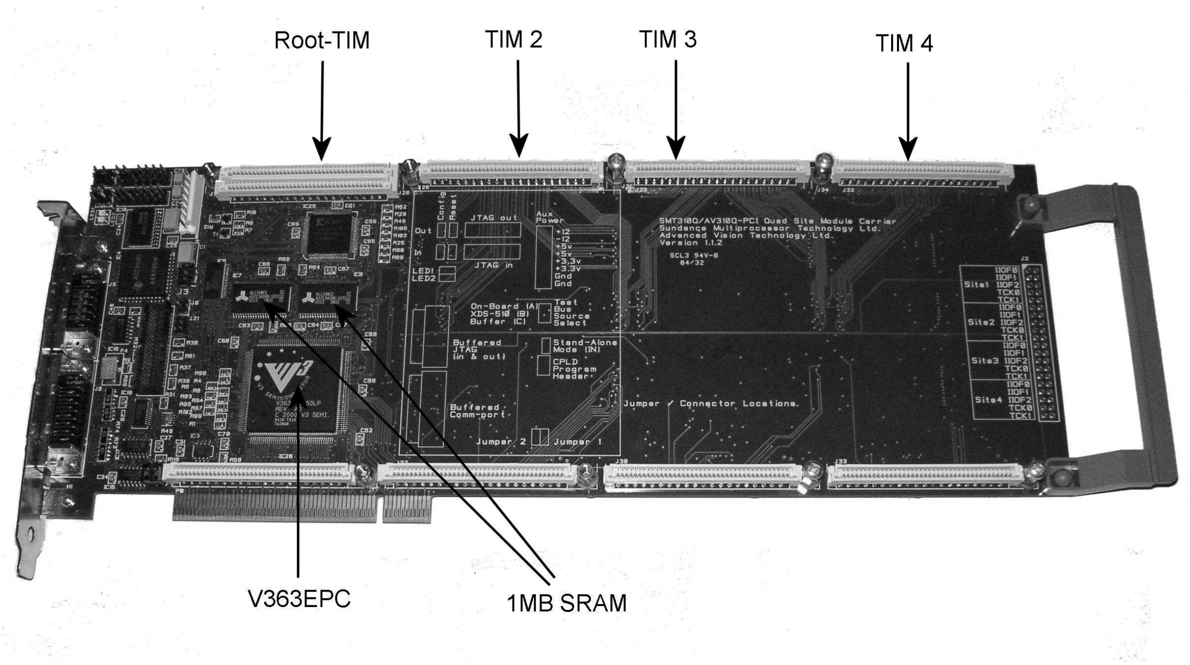


Figure C.1: Sundance® SMT310Q carrier board

Fig. C.2 shows the carrier board equipped with all used modules. The interconnection between the modules is performed with the high-speed bus (SHB).

The Sundance® SMT365 module [83], shown in Fig. C.3, is placed in the first TIM-slot and controls the operation of other modules. It consists principally of an FPGA, a DSP, and the high-speed bus-interfaces (SHB-interface, up to 400 MB/s) for fast data exchange with the other modules. In the prototype the DSP is used as a control device for the ADC/DAC and memory modules and to set the parameters for the pre-distortion filter running in real-time on the FPGA at the module SMT370.

The ADC/DAC-module SMT370 [81], see Fig. C.4, provides the interface to the analogue part of the prototype using high-speed ADC/DAC-converters. Further, the FPGA hosts the pre-distortion filter. Two high-speed SHB-interfaces, one of these being used for transferring the sampled output signal to the memory module SMT351-G, are also provided.

For storing the sampled output signal the module SMT351-G [82], shown in Fig. C.5, is used. This module provides 1 GB of fast memory, allowing to capture the sampled output signal. The interface with the other modules is the high-speed bus SHB (input from the ADC, the output is connected to the DSP-module SMT365).

---

[1] TIM stands for Texas Instruments Module and is an open standard designed to simplify the integration of multiple TMS320C4xs and other processors with communications ports into a system.

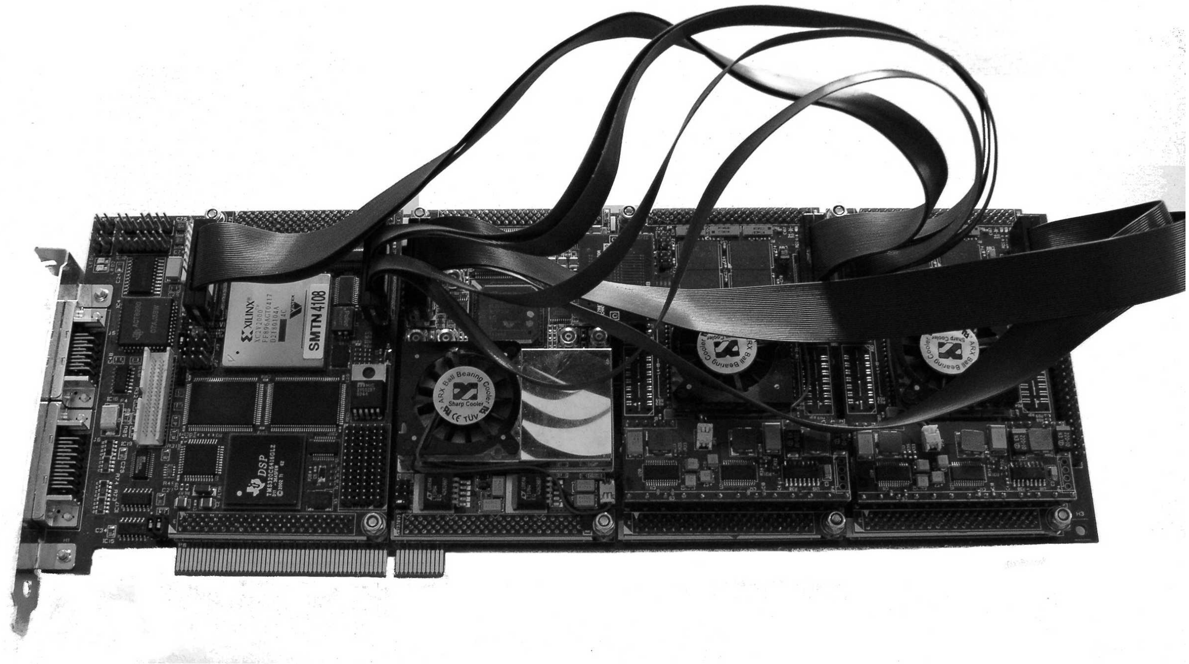


Figure C.2: Sundance® SMT310Q carrier board, equipped with all modules

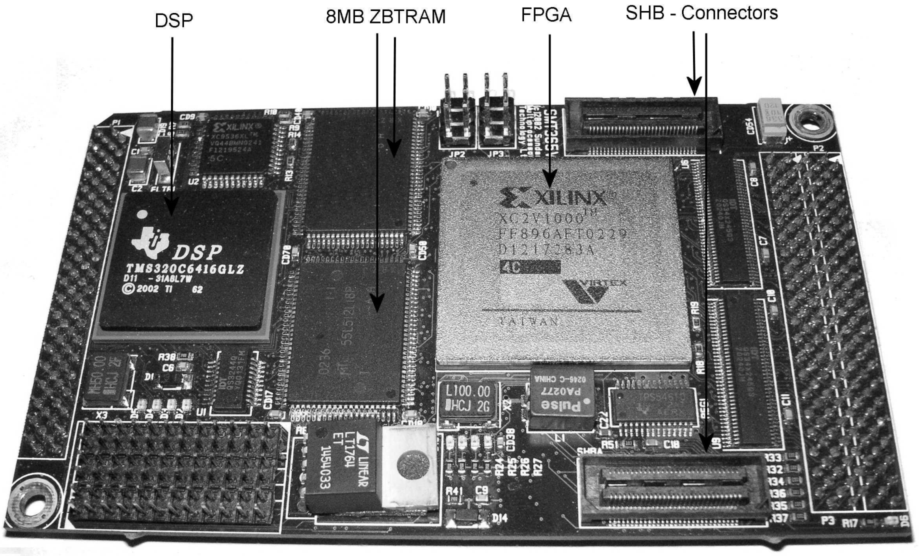


Figure C.3: Sundance® SMT365 DSP/FPGA-module

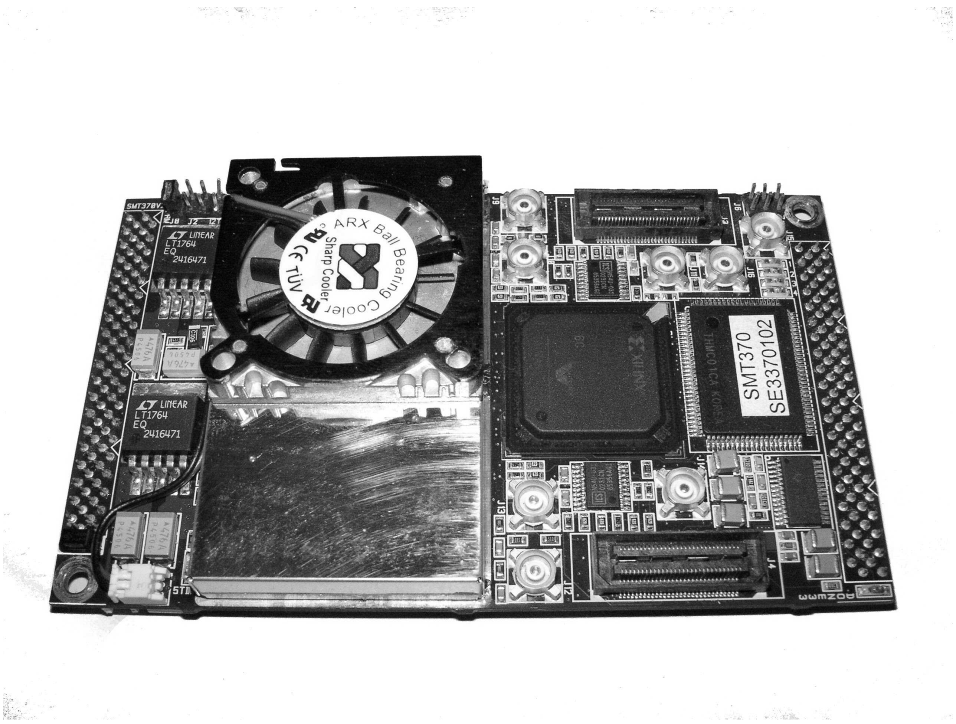


Figure C.4: Sundance® SMT370 ADC/DAC-module

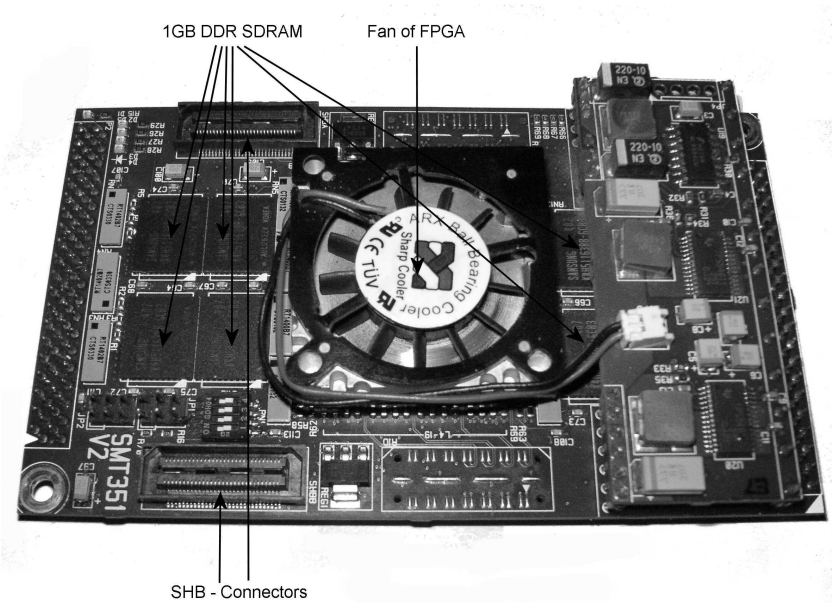


Figure C.5: Sundance® SMT351 memory-module

Fig. C.6 illustrates the RF-part of the experimental setup. The small amplifier (Minicircuits ZJL-4HG, in lower right corner) is used to boost the signal for the power amplifier (Minicircuits ZVE-8G, shown in the lower left corner).

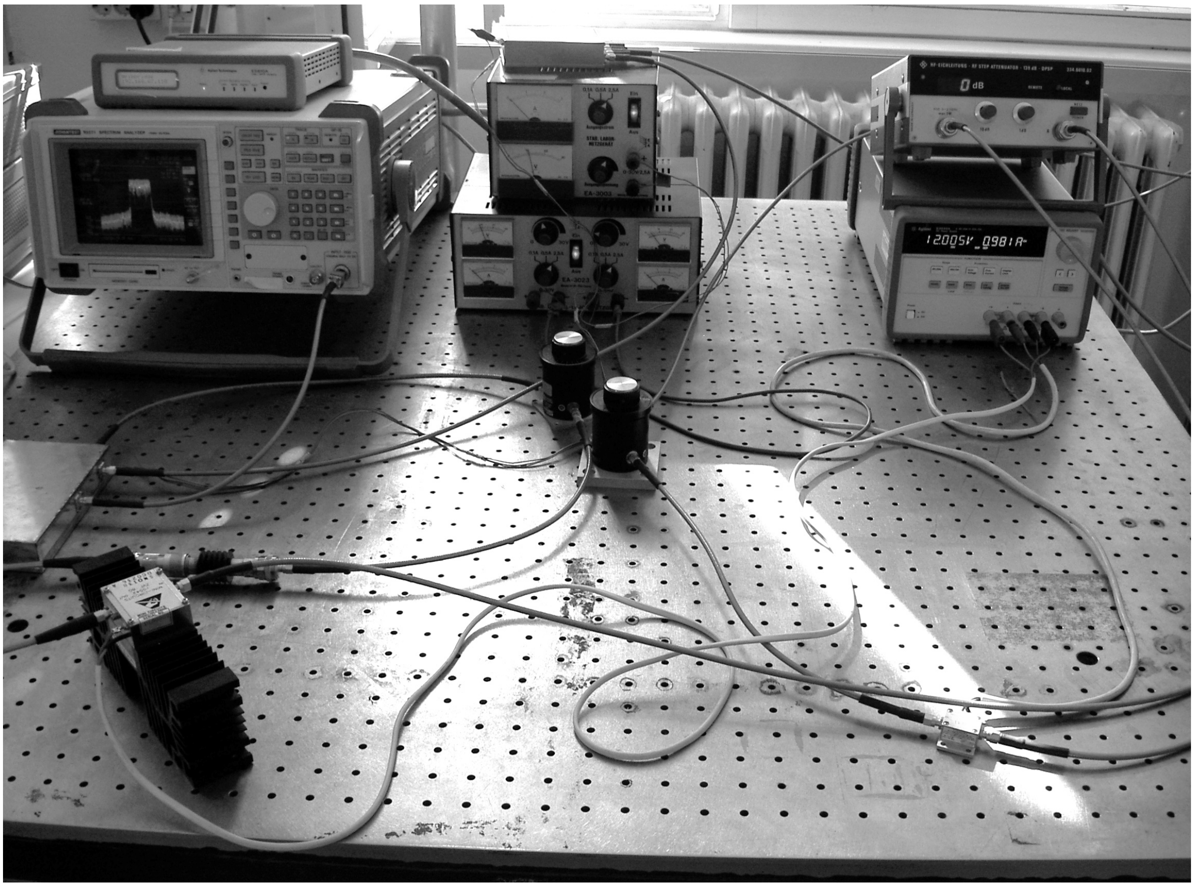


Figure C.6: RF-part of the experimental setup

Appendix D

Appendix: Abbreviations and Symbols

D.1 List of Abbreviations

ADS	advanced design system (Agilent Technologies)
AM-AM	amplitude-to-amplitude conversion
AM-PM	amplitude-to-phase conversion
ADC	analogue-to-digital converter
ASIC	application specific integrated circuit
BPF	bandpass filter
CDMA	code division multiple access
DAC	digital-to-analogue converter
DUC	digital up-conversion
DPD	digital pre-distortion
DSP	digital signal processor
EDGE	enhanced data rates for global evolution
ε-NLMS	epsilon-normalised least-mean-squares
FDD	frequency division duplex
FIR	finite impulse response
FPGA	field programmable gate array
GSM	global system for mobile communication
HPA	high-power amplifier
IBO	input power back-off
IF	intermediate frequency
ISM	industrial, scientific, medical
LMS	least-mean-squares
LDMOS	laterally diffused metal on silicon
NLMS	normalised least-mean-squares
OBO	output power back-off
PA	power amplifier
PC	personal computer
RBW	resolution bandwidth
RF	radio frequency

SDR	software defined radio
SNR	signal-to-noise ratio
VBW	video bandwidth
VHDL	very high speed integrated circuit hardware description language
UMTS	universal mobile telecommunications system
WCDMA	wideband code division multiple access
WLAN	wireless local area network

D.2 List of Mathematical Symbols

a, α	scalar constant
$x(t)$	function, parameter is t
$x[n]$	sequence, parameter is n
$X(f)$	Fourier-transform
$\mathbf{x}$	vector
$\mathbf{X}$	matrix
$\mathbb{V}$	operator
$\Re\{y\}$	real part of y
$\Im\{y\}$	imaginary part of y
$\boldsymbol{\theta}$	parameter vector
$J(\cdot)$	cost function
$J_r(\cdot)$	relative cost
$\mathscr{M}$	set
f	frequency
$\|\cdot\|$	norm
$\|\cdot\|_2$	ℓ_2-norm

Bibliography

[1] S. C. Cripps, *RF Power Amplifiers for Wireless Communications.* Artech House, 1999.

[2] P. B. Kenington, *High-Linearity RF Amplifier Design.* Artech House, 2000.

[3] A. A. M. Saleh and J. Salz, "Adaptive linearization of power amplifiers in digital radio systems," *The Bell System Technical Journal*, vol. 62, no. 4, pp. 1019–1033, Apr. 1983.

[4] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems.* John Wiley, 1980.

[5] W. J. Rugh, *Nonlinear System Theory, The Volterra/Wiener Approach.* The John Hopkins University Press, 1981.

[6] I. W. Sandberg, "Expansions for nonlinear systems," *The Bell System Technical Journal*, vol. 61, no. 2, pp. 159–199, Feb. 1982.

[7] S. Boyd, L. O. Chua, and C. A. Desoer, "Analytical foundations of Volterra series," *IMA Journal of Mathematical Control & Information*, vol. 1, pp. 243–282, 1984.

[8] S. Boyd and L. O. Chua, "Fading memory and the problem of approximating nonlinear operators with Volterra series," *IEEE Transactions on Circuits and Systems*, vol. CAS-32, no. 11, pp. 1150–1161, Nov. 1985.

[9] A. A. M. Saleh, "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Transactions on Communications*, vol. COM-29, no. 11, pp. 1715–1720, Nov. 1981.

[10] E. Aschbacher and M. Rupp, "Modelling and identification of a nonlinear power-amplifier with memory for nonlinear digital adaptive pre-distortion," in *Proc. IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC'03)*, pp. 555–559, June 2003.

[11] ——, "Identification of a nonlinear power-amplifier L-N-L structure for pre-distortion purposes," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'03)*, pp. 2102–2106, Sept. 2003.

[12] E. Aschbacher, H. Arthaber, and M.Rupp, "A fast algorithm for digital pre-distortion of nonlinear power amplifiers," in *Proc. of the 13th European Signal Processing Conference (EUSIPCO'05)*, Antalya, Turkey, Sept. 2005.

[13] A. H. Sayed and M. Rupp, "Error-energy bounds for adaptive gradient algorithms," *IEEE Transactions on Signal Processing*, vol. 44, no. 8, pp. 1982–1989, Aug. 1996.

[14] A. H. Sayed, *Fundamentals of Adaptive Filtering.* J. Wiley, 2003.

[15] E. Aschbacher and M. Rupp, "Robust identification of an L-N-L system," in *Conference Record of the 37th Asilomar Conference on Signals, Systems, and Computers (Asilomar'03)*, vol. 2, pp. 1298–1302, Nov. 2003.

[16] ——, "Robustness analysis of a gradient identification method for a nonlinear Wiener system," in *Proceedings of the 2005 IEEE Workshop on Statistical Signal Processing (SSP'05)*, July 2005.

[17] C. Eun and E. J. Powers, "A new Volterra predistorter based on the indirect learning architecture," *IEEE Transactions on Signal Processing*, vol. 45, no. 1, pp. 223–227, Jan. 1997.

[18] R. D. Nowak and B. D. V. Veen, "Volterra filter equalization: A fixed point approach," *IEEE Transactions on Signal Processing*, vol. 45, no. 2, pp. 377–387, Feb. 1997.

[19] E. Aschbacher, M. Steinmair, and M. Rupp, "Iterative linearization methods suited for digital pre-distortion of power amplifiers," in *Conf. Record of the 38th Asilomar Conf. on Signals, Systems, and Computers*, vol. 2, pp. 2198–2202, Nov. 2004.

[20] P. Brunmayr, "Implementation of a nonlinear digital pre-distortion algorithm," Master's thesis, Institute of Communications and Radio-Frequency Eng., Vienna University of Technology, 2005.

[21] M. Y. Cheong, E. Aschbacher, P. Brunmayr, M. Rupp, and T. I. Laakso, "Comparison and experimental verification of two low-complex digital predistortion methods," in *Conf. Record of the 39th Asilomar Conf. on Signals, Systems, and Computers*, Nov. 2005.

[22] 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Base Station (BS) radio transmission and reception (FDD) (Release 6), TS 25.104. [Online]. Available: http://www.3gpp.org

[23] Software Defined Radio Forum. [Online]. Available: http://www.sdrforum.org

[24] H. S. Black, "Translating system," *U.S. Patents 1.686.792 and 2.102.671*, Oct.,Dec. 1928,1937.

[25] H. Seidel, "A microwave feedforward experiment," *Bell Systems Technical Journal*, vol. 50, pp. 22 879–2916, Nov. 1971.

[26] K. J. Parsons and P. B. Kenington, "The efficiency of a feedforward amplifier with delay loss," *IEEE Transactions on Vehicular Technology*, vol. 43, no. 2, pp. 407–412, May 1994.

[27] ——, "Effect of delay mismatch on a feedforward amplifier," *IEE Proceedings on Circuits, Devices, and Systems*, vol. 141, no. 2, pp. 140–144, April 1994.

[28] V. Petrovic and C. N. Smith, "Reduction of intermodulation distortion by means of modulation feedback," in *Proceedings of the IEE Conference on Radio Spectrum Conservation Techniques*, pp. 44–49, Sept. 1983.

[29] A. Bateman and D. M. Haines, "Direct conversion tranceiver design for compact low cost portable mobile terminals," in *Proceedings of the 39th IEEE Vehicular Technology Conference*, vol. 1, pp. 57–62, May 1989.

[30] M. Johanson and T. Mattson, "Linearised high-efficiency power amplifier for PCN," *Electronics Letters*, vol. 27, no. 9, pp. 762–764, 1991.

[31] P. B. Kenington, R. J. Wilkinson, and K. J. Parsons, "Noise performance of a cartesian loop transmitter," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 2, pp. 467–476, May 1997.

[32] E. Biglieri, S. Barberis, and M. Catena, "Analysis and compensation of nonlinearities in digital transmission systems," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 1, pp. 42–51, Jan. 1988.

[33] G. Karam and H. Sari, "Improved data predistortion using intersymbol interpolation," in *Proceedings of the IEEE International Conference on Communications*, vol. 1, pp. 286–291, June 1989.

[34] S. Pupolin, A. Sarti, and H. Fu, "Performance analysis of digital radio links with nonlinear transmit amplifier and data predistorter with memory," in *Proceedings of the IEEE International Conference on Communications*, vol. 1, pp. 292–296, June 1989.

[35] G. Lazzarin, S. Pupolin, and A. Sarti, "Nonlinearity compensation in digital radio systems," *IEEE Transactions on Communications*, vol. 42, no. 2/3/4, pp. 988–999, Feb./March/April 1994.

[36] M. Schetzen, "Theory of pth-order inverses of nonlinear systems," *IEEE Transactions on Circuits and Systems*, vol. CAS-23, no. 5, pp. 285–291, May 1976.

[37] S. Benedetto and E. Biglieri, "Nonlinear equalization of digital satellite channels," *IEEE Journal on Selected Areas in Communications*, vol. SAC-1, no. 1, pp. 57–62, Jan. 1983.

[38] A. Bateman, D. M. Haines, and R. J. Wilkinson, "Linear transceiver architectures," in *Proceedings of the 38th IEEE Vehicular Technology Conference*, pp. 478–484, May 1988.

[39] Y. Nagata, "Linear amplification technique for digital mobile communication," in *Proceedings of the 39th IEEE Vehicular Technology Conference*, vol. 1, pp. 159–164, May 1989.

[40] J. Cavers, "Amplifier linearization using a digital predistorter with fast adaptation and low memory requirements," *IEEE Transactions on Vehicular Technology*, vol. 39, no. 4, pp. 374–382, Nov. 1990.

[41] J. Kim and K. Konstantinou, "Digital predistortion of wideband signals based on power amplifier model with memory," *Electron. Lett.*, vol. 37, no. 23, pp. 1417–1418, Nov. 2001.

[42] J. W. Wustenberg, H. J. Xing, and J. R. Cruz, "Complex gain and fixed-point predistorters for CDMA power amplifiers," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 2, pp. 469–478, March 2004.

[43] M. Y. Cheong, S. Werner, J. Cousseau, and T. I. Laakso, "Predistorter identification using the simplicial canonical piecewise linear function," in *Proc. of the 12th International Conference on Telecommunications ICT2005*, Cape Town, South Africa, pp. 2102–2106, May 2005.

[44] L. Ljung, *System Identification, Theory for the User*, 2nd ed. Prentice Hall, 1999.

[45] V. Volterra, *Theory of Functions and of Integral and Integro-Differential Equations.* New York: Dover Publ., 1959.

[46] Advanced Design System (ADS), Agilent Technologies. [Online]. Available: http://eesof.tm.agilent.com/products/ads_main.html

[47] N. Wiener, *Nonlinear Problems in Random Theory.* John Wiley, 1958.

[48] D. G. Luenberger, *Optimization by Vector Space Methods.* J. Wiley, 1968.

[49] J. A. Dieudonné, *Foundations of Modern Analysis.* Academic Press, 1967.

[50] D. R. Hummels and R. D. Gitchell, "Equivalent low-pass representations for bandpass volterra systems," *IEEE Transactions on Communications*, vol. COM-28, no. 1, pp. 140–142, Jan. 1980.

[51] J. I. Churkin, C. P. Jakowlew und G. Wunsch, *Theorie und Anwendung der Signalabtastung.* VEB Verlag Technik Berlin, 1966.

[52] M. Unser, "Sampling - 50 years after Shannon," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 569–587, April 2000.

[53] R. J. Martin, "Volterra system identification and Kramer's sampling theorem," *IEEE Transactions on Signal Processing*, vol. 47, no. 11, pp. 3152–3155, Nov. 1999.

[54] Y.-M. Zhu, "Generalized sampling theorem," *IEEE Transactions on Circuits and Systems—Part II: Analog and Digital Signal Processing*, vol. 39, no. 8, pp. 587–588, Aug. 1992.

[55] J. Tsimbinos and K. V. Lever, "Input Nyquist sampling suffices to identify and compensate nonlinear systems," *IEEE Transactions on Signal Processing*, vol. 46, no. 10, pp. 2833–2837, Oct. 1998.

[56] H. Akaike, "Fitting autoregressive models for prediction," *Annals of the Institute of Statistical Mathematics*, vol. 21, pp. 243–247, 1969.

[57] J. Rissanen, "Modelling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, Sept. 1978.

[58] Mini-Circuits Medium to High Power Amplifiers: Data sheet. [Online]. Available: http://www.minicircuits.com/dg03-176.pdf

[59] S. Boyd, "Multitone signals with low crest factor," *IEEE Transactions on Circuits and Systems*, vol. CAS-33, no. 10, pp. 1018–1022, Oct. 1986.

[60] Rohde&Schwarz. [Online]. Available: http://www.rohde-schwarz.com/

[61] Agilent Technologies. [Online]. Available: http://www.agilent.com/

[62] Sundance Multiprocessor Technology Ltd. [Online]. Available: http://www.sundance.com

[63] R. Langwieser, "Entwicklung von HF-Baugruppen für ein Echtzeit MIMO-Übertragungssystem," Master's thesis, Institut für Nachrichten- und Hochfrequenztechnik, Technische Universität Wien, 2004.

[64] N. J. Bershad, S. Bouchired, and F. Castanie, "Stochastic analysis of adaptive gradient identification of Wiener-Hammerstein systems for gaussian inputs," *IEEE Transactions on Signal Processing*, vol. 48, no. 2, pp. 557–560, Feb. 2000.

[65] N. J. Bershad, P. Celka, and S. McLaughlin, "Analysis of stochastic gradient identification of Wiener-Hammerstein systems for nonlinearities with Hermite polynomial expansions," *IEEE Transactions on Signal Processing*, vol. 49, no. 5, pp. 1060–1072, May 2001.

[66] A. D. Kalafatis, L. Wang, and W. Cluett, "Identification of Wiener-type nonlinear systems in a noisy environment," *International Journal of Control*, vol. 66, no. 6, pp. 923–941, April 1997.

[67] M. Rupp and A. H. Sayed, "Supervised learning of perceptron and output feedback dynamic networks: A feedback analysis via the small gain theorem," *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 612–622, May 1997.

[68] S. Haykin, *Adaptive Filter Theory*, 4th ed. Prentice Hall, 2002.

[69] H. K. Khalil, *Nonlinear Systems.* Macmillan Pub., 1992.

[70] S. Boyd and L. O. Chua, "Uniquiness of a basic nonlinear structure," *IEEE Transactions on Circuits and Systems*, vol. CAS-30, no. 9, pp. 648–651, Sept. 1983.

[71] M. Rupp and A. H. Sayed, "Robustness of Gauss-Newton recursive methods: A deterministic feedback analysis," *Signal Processing*, vol. 50, pp. 165–187, 1996.

[72] K. J. Muhonen, M. Kavehrad, and R. Krishnamoorthy, "Look-up table techniques for adaptive digital predistortion: A development and comparison," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 5, pp. 1995–2002, Sept. 2000.

[73] P. Andreani, L. Sundstrom, N. Karlsson, and M. Svensson, "A chip for linearization of RF power amplifiers using digital predistortion with a bit-parallel complex multiplier," in *Proc. of the1999 IEEE Intern. Symp. on Circuits and Systems (ISCAS '99)*, vol. 1, pp. 346–349, Mai 1999.

[74] M. Steinmair, "Identifikation und Linearisierung nichtlinearer Verstärker mit Volterrareihen," Master's thesis, Insitut für Nachrichten- und Hochfrequenztechnik, Technische Universität Wien, 2004.

[75] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, 1992. [Online]. Available: http://www.nr.com

[76] C. G. Broyden, "A class of methods for solving nonlinear simultaneous equations," *Mathematics of Computation*, vol. 19, no. 92, pp. 577–593, Oct. 1965.

[77] XILINX Virtex-II Platform FPGAs: Complete data sheet. [Online]. Available: http://www.xilinx.com/partinfo/ds031.pdf

[78] TMS320C64xx fixed-point Digital Signal Processors: Complete data sheet. [Online]. Available: http://focus.ti.com/lit/ds/symlink/tms320c6416.pdf

[79] Analog Devices D/A converter AD9777: Complete data sheet. [Online]. Available: http://www.analog.com/UploadedFiles/Data_Sheets/3229938536490156500AD9777_b.pdf

[80] Analog Devices A/D converter AD6645: Complete data sheet. [Online]. Available: http://www.analog.com/UploadedFiles/Data_Sheets/39459366394038040210399061519954AD6645_b_.pdf

[81] Sundance SMT370-AC module: Data sheet. [Online]. Available: http://www.sundance.com/edge/files/productpage.asp?STRFilter=SMT370-AC

[82] Sundance SMT351-G module: Data sheet. [Online]. Available: http://www.sundance.com/edge/files/productpage.asp?STRFilter=SMT351-VP7-5-G

[83] Sundance SMT365-16-2 module: Data sheet. [Online]. Available: http://www.sundance.com/edge/files/productpage.asp?STRFilter=SMT365-16-2

[84] Mini-Circuits Low to Medium Power Amplifiers: Data sheet. [Online]. Available: http://www.minicircuits.com/dg03-174.pdf

[85] XILINX IP Core Pipelined Divider: Data Sheet. [Online]. Available: http://www.xilinx.com/bvdocs/ipcenter/data_sheet/sdivider.pdf

[86] G. Doblinger, *Signalprozessoren - Architekturen, Algorithmen, Anwendungen.* J. Schlembach Fachverlag, 2000.

[87] Mentor Graphics. [Online]. Available: http://www.mentor.com

[88] Sundance SMT310Q PCI Carrier Board: Data sheet. [Online]. Available: http://www.sundance.com/docs/SMT310Q%20User%20Manual.pdf

Curriculum Vitae

Personal Details

Name	Ernst Aschbacher
Marital status	Single
Citizenship	Italian
Date of birth	04.09.1970

Education

09/1984 – 06/1986	Full-time vocational school for Electrical Engineering, Bruneck (IT)
09/1987 – 06/1990	Technical college (Gewerbeoberschule), Bozen (IT) Branch: Informatics
09/1991 – 07/1993	Technical college (Gewerbeoberschule), Bozen (IT) Branch: Informatics
09/1993 – 06/2000	Master studies at Vienna University of Technology (AUT) Electrical Engineering and Information Technology
06/2000	Graduation with highest honours (Diplomingenieur)
09/2001 – 09/2005	Ph.D. studies at Vienna University of Technology (AUT) Field: Signal Processing for Mobile Communications

Military Service

09/1990 – 09/1991	Military service in Meran (IT), military music

Professional Experience

07/1986 – 09/1990	Full-time millwright, GKN Birfield, Bruneck (IT)
10/2000 – 09/2001	Project assistant at the Institute of Communications and Radio-Frequency Eng. Vienna University of Technology Field: Optical Communications
09/2001 – 09/2005	Assistant (Universitätsassistent) at the Institute of Communications and Radio-Frequency Eng. Vienna University of Technology

Language Skills

- German (mother tongue)
- English (business fluent)
- Italian (business fluent)
- Spanish (basics)