

CHALMERS



Comparison of digital predistortion identification methods for RF power amplifiers

Master of Science Thesis in the Master Degree Programme, Systems, Control and Mechatronics

ANNIKA ESTHER MAYER

Department of Signals and Systems
Division of Communication Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden, 2013
Master's Thesis 2013:xx

Comparison of digital predistortion identification methods for RF power amplifiers
ANNIKA E. MAYER

@ ANNIKA E. MAYER, 2013.

Master's Thesis 2013:xx
Department of Signals and Systems
Division of Communication Engineering
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Email annika.est.mayer@gmail.com



University of Stuttgart
Germany



CHALMERS

REPORT NO. 2013/XXXX

Comparison of digital predistortion identification methods for RF power amplifiers

ANNIKA ESTHER MAYER

Department of Signals and Systems
Division of Communication Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden, 2013

Abstract

Radio frequency power amplifiers (PA) are a part of wireless transmitters. They amplify the signal before it is transmitted. Unfortunately efficient power amplifiers are inherently nonlinear. The nonlinearity of the power amplifier distorts the signal. Therefore the nonlinearity has to be compensated with a predistortion. A computational efficient and wide-spread method to identify the predistortion coefficients is indirect learning architecture (ILA). It can be shown that the coefficients identified with indirect learning architecture are not identical to the optimal coefficients. They are biased due to the input noise bias problem and due to the cascade order problem. In this thesis a new method based on approximating a post-replica nonlinear predistortion problem by a weighted least-squares problem is suggested in order to avoid the cascade order problem. Furthermore a model-based approach mitigating the noise bias problem is further investigated. The focus of the investigation lies on better understanding the trade-off between reducing the noise bias and introducing an additional modeling error.

Acknowledgements

First of all I want to thank Thomas Eriksson for giving me the opportunity to write this thesis, for his guidance, his time and his constructive support. I also want to acknowledge Per Landin and thank him for his assistance throughout this work and for giving me great feedback. Many thanks to Daniella Schittler and Frank Allgöwer for co-supervising my thesis in Stuttgart. Furthermore I want to thank Camille Janvier who traveled to Chalmers from Paris to oppose my thesis. And finally many thanks to Christian Fager for suggesting the topic to me.

I also want to express my gratitude to all those people who made the double degree program between Chalmers and the University of Stuttgart possible and who helped me with organizational issues. Especially to Georg Seyboth, Marcus Reble, Beate Spinner, Claudia Vetter and Frank Allgöwer from the Institute for Systems Theory and Automatic Control, Marion Echle from the international affairs office, Knut Åkesson from the department of signals and systems, Marie Iwanow the director of studies for the systems, control and mechatronics master program and Anders Fréden, the double master degree coordinator from Chalmers. I actually don't know how much work and effort they have put into setting up this exchange program, because everything was so well organized and went so smoothly. Which is an even bigger compliment to all of them as we were the first generation of students from Stuttgart taking part in this exchange.

Further I want to thank the DAAD (Deutscher Akademischer Austauschdienst) and my father for their financial support.

I want to thank all the great people I have met in Sweden, for making my stay so enjoyable. I can not mention them all but hope they know whom I mean. Especially I want to thank Lars for making me discover the secrets of Swedish pastry, Alexandra for the awesome time we spent ice skating, swimming, cycling and relaxing with French comedies afterwards, Catarina and Camille for our delicious cooking evenings and the most in-depth wine discussion I ever had, Khartik and Mannen for not pushing me out of the canoe, even though I was constantly “s(th)inking”. Moreover I want to thank the best-ever-Indian-Italian volleyball team, the Frolunda gang, the Wireless class, my fantastic kitchen mates, the Chalmers Sailing Club, my Swedish class mates at Folksuniversität, the Gothenburg Outdoor Hockey Tigers and the Valhalla Landhockey Club.

Last but not least I want to thank my family. Especially my mother, my father and my sister. Thanks for always being there for me with your support and your guidance, with your mind and your heart, with your tears and your smiles. I wish you all the best in live.

Contents

1	Introduction	12
1.1	A communication system	12
1.1.1	Examples for communication systems	12
1.1.2	A Shannon-style block diagram	12
1.1.3	Electromagnetic waves	12
1.1.4	Coding, modulation. A brief discussion	13
1.1.5	Frequency channels	15
1.1.6	Frequency allocation	15
1.2	The analog-front end	16
1.2.1	Microwave components	17
1.2.2	Traditional receiver and transmitter architecture	17
1.2.3	Modern transmitter architecture	18
1.3	Problem description	18
2	Amplifier modeling	21
2.1	Amplifier characteristics	21
2.2	Amplifier models	21
2.2.1	Parametric memoryless power amplifier models	22
2.2.2	Parametric memorybased power amplifier models	23
2.2.3	Linear basis function model	23
2.3	Model identification	25
2.4	Performance metrics	25
3	Digital predistortion (DPD)	27
3.1	The general idea of DPD	27
3.2	Mathematical formulation of the DPD identification problem	27
3.3	Digital predistortion identification with the indirect learning architecture (ILA)	29
3.3.1	Predistortion versus postdistortion	29
3.3.2	Indirect learning architecture	30
3.3.3	Obtaining a good starting point for iterative ILA	31
4	Discussion and analysis of problems regarding ILA	33
4.1	Extrapolation problem	33
4.2	Cascade order problem	33
4.3	Noise bias problem	34
5	Proposed solutions	36
5.1	Reinforcement data grid	36
5.2	Weighted indirect learning architecture (WILA)	37
5.2.1	Algorithm	37
5.2.2	Derivation	38
5.2.3	Magnitude model identification	41
5.3	Model-based indirect learning architecture (MILA)	41
5.3.1	Algorithm	42
5.3.2	Derivation, error and model identification	42
6	Results and Discussion	44
6.1	Illustrating example	44
6.1.1	Suggestion 1: Reinforcement data grid for the initial guess	44
6.1.2	Suggestion 2: Weighted indirect learning architecture	45
6.2	Testbench	46
6.2.1	Suggestion 1: Reinforcement data grid for the initial guess	47
6.2.2	Suggestion 2: Weighted indirect learning architecture	49
6.2.3	Suggestion 3: Model-based indirect learning architecture	50
6.3	Summary conclusion	51

7 Future work	52
Appendices	53
A Probability density distributions	53
B Approximating an integral by a sum of a statistical variable	53
C Transformation properties of probability density distributions	53
D Simulation setup and simulation parameters	54

List of Figures

1	Basic architecture of a wireless communication system	12
2	Farfield approximation	12
3	Linear and circular polarization of an electromagnetic plane wave	13
4	Waveforms ASK, FSK and PSK [4]	13
5	Constellation diagram 8-PSK and 16-QAM	14
6	Pulse shaping filter	14
7	Probability density functions for a complex Gaussian random signal	15
8	Frequency channel and adjacent channels	15
9	Frequency allocation plan US [5]	16
10	Symbols for common microwave components	17
11	Block diagram: Receiver	18
12	Block diagram: Transmitter	18
13	Block diagram: Modern transmitter architecture	19
14	PA measurement data and confidence intervals. The data is normalized and was identified with a multitone signal with a relative bandwidth of 0.1.	21
15	Normalized AM/AM (amplitude-to-magnitude) and AM/PM (phase-to-magnitude) plots for memoryless Arctan, Saleh and Ghorbani power amplifier model.	22
16	Adjacent channel power (ACPR)	26
17	Block diagram: Digital predistortion	27
18	Block diagram predistortion and postdistortion	29
19	Comparison convex and non-convex optimization problem	30
20	Postdistortion for ILA	31
21	Block diagram: Indirect learning architecture (i.e. [13] Fig.1 or [23] Fig.6)	32
22	Histogram for the PA input magnitude $ u $ with and without a predistortion	32
23	Input and output data distribution seen by the PA for the predistorted and the undistorted signal, with an decreased input power	33
24	Input and output data distribution seen by the PA for the predistorted and the undistorted signal, with an adjusted linear gain	34
25	Weighted polynomial fitting and weighted squared-errors	35
26	Weight, Function and we^2 for an ILA solution	35
27	Input noise bias	36
28	Bias due to additive input noise	36
29	Extrapolation problem without border area reinforcement for the model identification	37
30	Comparison predistortion optimization problem, postdistortion optimization problem (ILA) and postreplica optimization problem (WILA)	38
31	Comparison convergence of the predistortion function with and without reinforcement data grid for a backoff of 14.37 dB	44
32	Comparison convergence of the predistortion function with and without reinforcement data grid for a backoff of 10.42 dB. The first estimate was underestimated.	44
33	Comparison convergence of the predistortion function with and without reinforcement data grid for a backoff of 10.42 dB. The first estimate was overestimated.	45
34	Weight, Function and we^2 for an WILA solution	45
35	Comparison ideal predistortion and estimated predistortion squared-error for ILA and WILA, backoff 14.37 dB	46
36	Comparison ideal predistortion and estimated predistortion squared-error for ILA and WILA, backoff 9.03 dB	46
37	Comparison ideal predistortion and estimated predistortion squared-error for ILA and WILA, backoff 8.59 dB	46
38	Illustration noise levels and maximum output power levels	47
39	ILA: Initial guess with and without reinforcement grid, SNR between 49.34 and 51.45 dB, PSNR 60 dB	47
40	ILA: Initial guess with and without reinforcement grid, SNR between 29.30 and 31.48 dB, PSNR 40 dB	48
41	WILA: Initial guess with and without reinforcement grid, SNR between 49.34 and 51.45 dB, PSNR 60 dB	48

42	WILA: Initial guess with and without reinforcement grid, SNR between 29.30 and 31.48 dB, PSNR 40 dB	48
43	Comparison ILA and WILA, NMSE for different noise levels and linearization areas . . .	49
44	Comparison ILA and WILA, ACPR for different noise levels and linearization areas . . .	49
45	Comparison MILA and ILA/WILA, NMSE for different noise levels and linearization areas	50
46	Comparison MILA and ILA/WILA, ACPR for different noise levels and linearization areas	50
47	Error composition MILA, squared error, model identification error and predistortion identification error	51
48	Memoryless simulation setup	54

List of Tables

1	Arctan model parameters [23]	22
2	Saleh model parameters [25]	23
3	Ghorbani model parameters [19]	23
4	Parametric memorybased PA models	24
5	Memory order and polynomial order for memory polynomial models	24
6	Simulation parameters	55
7	Substituted simulation parameters	55

1 Introduction

1.1 A communication system

In our every day lives, we are surrounded by wireless communication systems. The goal of this section is to impart the reader with a basic understanding on how those communication system work.

1.1.1 Examples for communication systems

Communication systems are an integral part of modern society. Imagine the daily routine of Joe Bloggs. His radio alarm clock wakes him up at 6.40 am. It receives the time information from the closest radio station. He goes to the kitchen and turns on his television. It's not loud enough, so he boosts the volume with a remote control. Driving to a business partner he uses his route guidance system, which receives GPS information from at least three satellites. On his way he makes a phone call to a friend. After work he returns home and opens his door with an RFID key. Then he accesses the internet via a WLAN network to check some emails. In each sentence Joe Bloggs used a wireless communication system. The applications are varied, but the basic principles and components of those wireless communication systems are the same.

1.1.2 A Shannon-style block diagram

Figure 1 shows a block diagram of a wireless communication system. Each communication link consists of an information source, a transmitter, a receiver and an information destination. Between transmitter and receiver the signal is transmitted in form of an electromagnetic wave. The received signal is disturbed by noise. There are several noise sources, for example adjacent channel noise or noise from non ideal physical components (i.e. antenna, power amplifier, ...) .

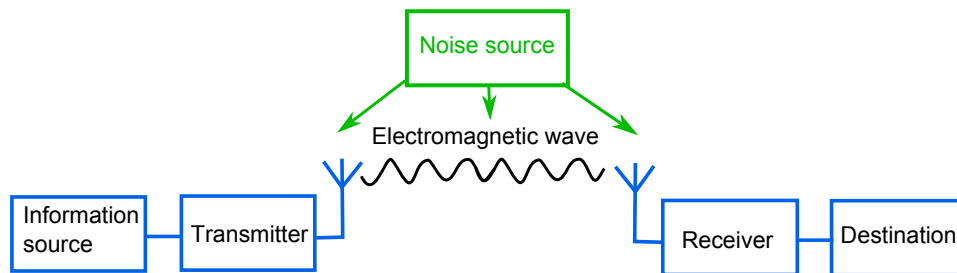


Figure 1: Basic architecture of a wireless communication system

1.1.3 Electromagnetic waves

An electromagnetic wave consists of a self-propagating perpendicular electrical and magnetical field. In the farfield, when the wave is far enough from the transmitting antenna, the wave can be approximated by a plane wave. For a plane wave surfaces with constant phase span parallel planes. Figure 2 illustrates the farfield plane wave formation. An electromagnetic plane wave can have three different polarizations,

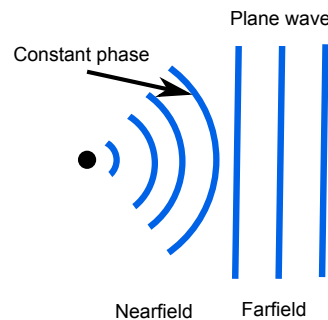


Figure 2: Farfield approximation

it can either be linearly polarized, right-handed circularly polarized or left-handed circularly polarized. For a linearly polarized plane wave the magnetic and electric field vectors oscillate in two perpendicular fixed planes, see Figure 3 (a) . For the circularly polarized electromagnetic waves the oscillation direction rotates in space, see Figure 3 (b) .

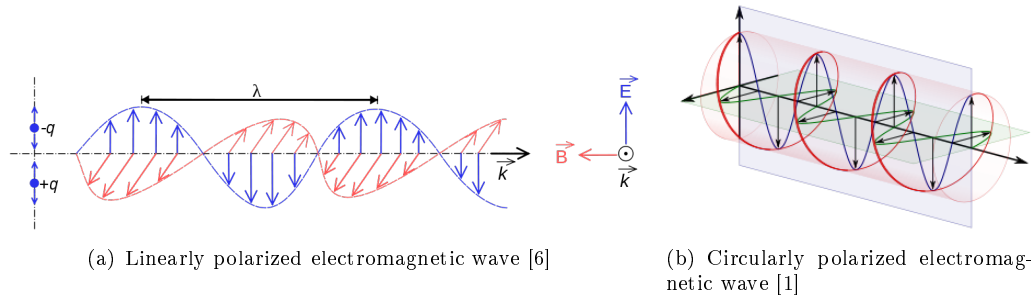


Figure 3: Linear and circular polarization of an electromagnetic plane wave

1.1.4 Coding, modulation. A brief discussion

The polarization is irrelevant, for understanding how information can be coded and transmitted with electromagnetic plane waves, because for each polarization the magnetic field and the electrical field are completely determined by each other. This means the information can be coded in only one of them. An electromagnetic plane wave has the following attributes, which can be used to transmit information:

- Frequency
- Phase
- Amplitude

The simple corresponding modulation schemes are frequency shift keying (FSK), phase shift keying (PSK) and amplitude shift keying (ASK). To transmit a 0 or 1 either the frequency, the phase or the amplitude is switched between two values, see Figure 4.

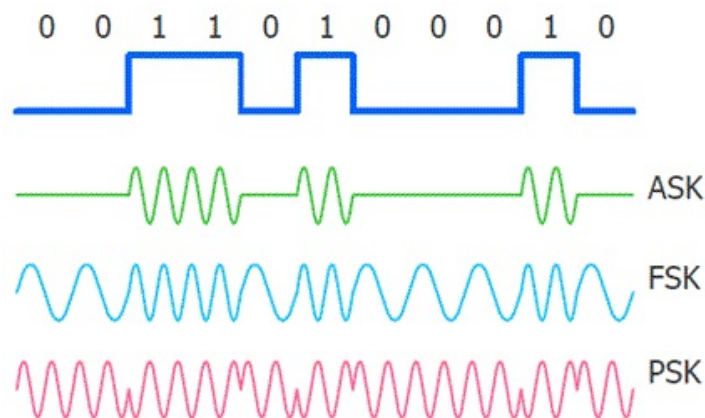


Figure 4: Waveforms ASK, FSK and PSK [4]

An important quantity to assess the quality of a modulation scheme is the bit error rate. The bit error rate describes the probability that a bit is detected wrongly. Besides the signal-to-noise ratio and the channel bandwidth, the modulation scheme determines the bit error rate. Amplitude-shift-keying for example, has a higher bit-error-rate than phase-shift-keying.

M-ary digital modulation More than one bit can be transmitted per signal interval, by using multiple discrete amplitude and phase levels. A figure that visualizes the amplitude and phase levels and the corresponding digital signature in the complex plane is called a constellation diagram. As an example the constellation diagrams for phase shift keying (PSK) and Quadrature Amplitude Modulation (QAM) are shown in Figure 5.

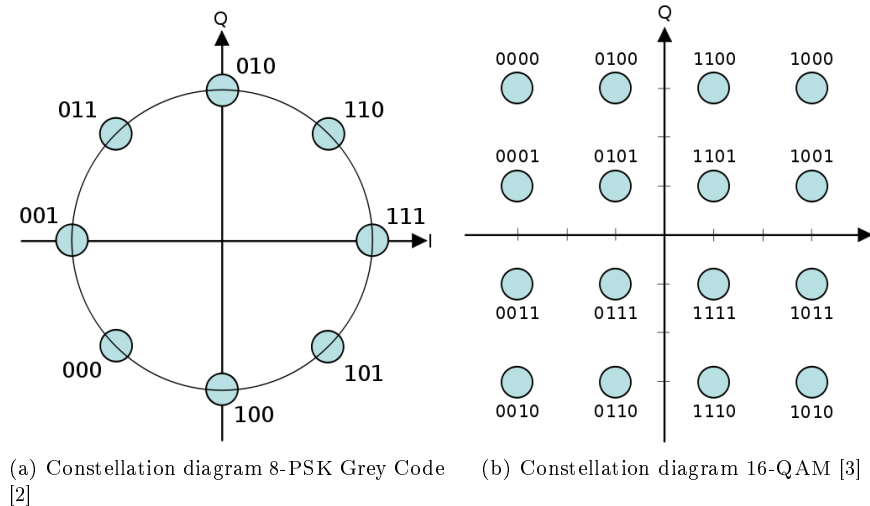


Figure 5: Constellation diagram 8-PSK and 16-QAM

The sharp phase transitions generate a signal with a large bandwidth. The bandwidth is reduced with a so called pulse-shaping filter. Figure 6 illustrates the effect of a pulse-shaping filter in the constellation diagram.

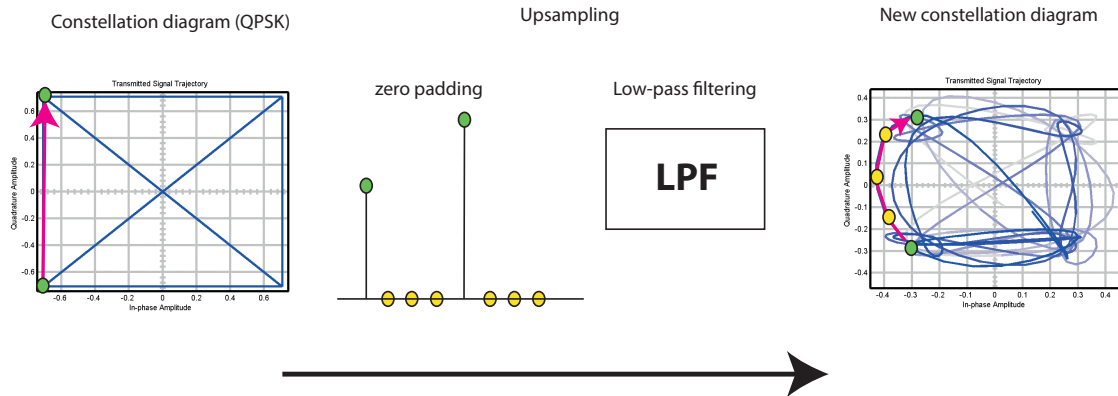


Figure 6: Pulse shaping filter

Multiplexing Multiplexing means that more than one user can transmit information in one channel at the same time. This is for example important for base-stations, where all mobile phone users transmit and receive signals in the same frequency channel. Without going into detail, it can be noted, that for several wide-spread multiplexing modulation schemes, the transmitted signal can be modeled as pulse-shape filtered complex Gaussian random process with zero mean as shown in [11], [18]. The probability density functions of the amplitude and phase of a complex Gaussian signal are Rayleigh-distributed and uniformly-distributed in the interval $[-\pi, \pi]$ respectively (see Figure 7 and Appendix A).

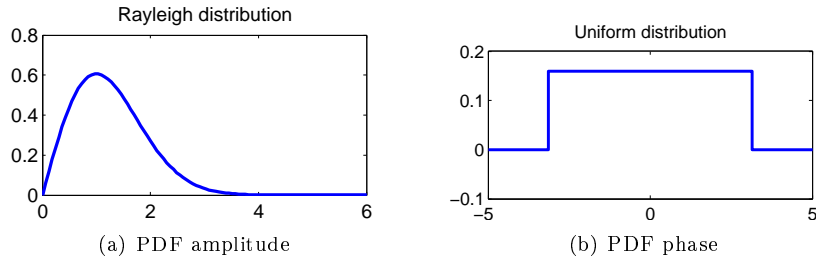


Figure 7: Probability density functions for a complex Gaussian random signal

1.1.5 Frequency channels

A channel is a frequency band which is allocated to a specific application. A frequency channel is characterized by the frequency band, or respectively the center frequency and the bandwidth. Each channel has two adjacent channels. The channels are separated by a channel gap. The adjacent channel is relevant, because it can leak. This means that the signal is not only transmitted in the channel but also in the adjacent channel. The signal leakage of the adjacent channel disturbs the signal. The signal leakage is especially severe if the transmission power of the adjacent channel is large compared to the signal power of the transmission channel.

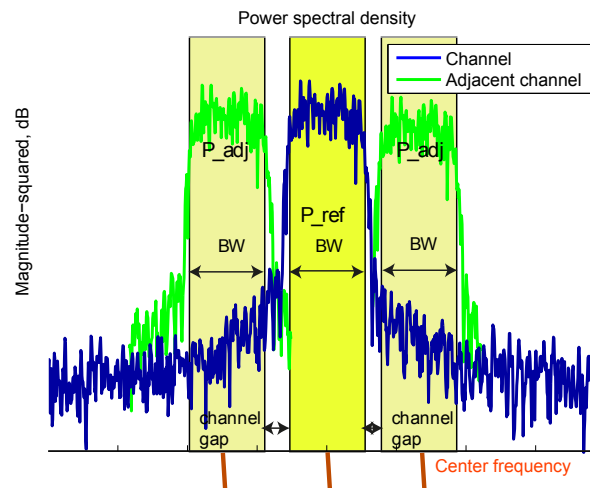


Figure 8: Frequency channel and adjacent channels

1.1.6 Frequency allocation

Electromagnetic waves are the medium of choice whenever information is transmitted wireless because they are literally „fast as light “. The property which distinguishes the electromagnetic signals used for transmitting the RFID and GPS information is the center frequency. The GPS information is transmitted at a frequency of 1575.42 MHz and 1227.6 MHz while the RFID information to open the entrance door is transmitted in the frequency band between 13.553 and 13.567 MHz. Other well known electromagnetic waves are visible light, whose frequency band ranges from 400 to 800 THz and X-rays whose frequency band ranges from 30 petahertz to 30 exahertz. For different applications, different frequencies are more or less useful. A frequency- dependent property is for example the antenna size, the lower the frequency the larger the antenna needs to be. Collecting data from the beginning of the universe takes place at lower frequencies than mobile phone communication, and visible light has a even higher frequency. That’s why an electromagnetic observatory is visible from far away, while a mobile phone fits in the trouser pocket, and millions of cone cells are located in everyman’s eyes. Another frequency-dependent property is the photon energy, the lower the frequency the smaller the energy. That’s the reason why X-rays and microwaves can interact with atoms and molecules, while

mobile phone and car key electromagnetic waves don't. The photons have less energy, and therefore the electromagnetic waves used for wireless communication are not dangerous for the human body. The technical accessible frequency spectrum is limited, at one end by impractical antenna sizes, and at the other end by miniaturization limits. Currently there are still harmless frequencies between 30 and 300 GHz, so called millimeter waves, which are not used for commercial data communication. Several research projects aim at developing them. The frequency spectrum is therefore a scarce resource. In each country the usage of different frequencies is regulated by so-called frequency allocation plans. Figure 9 shows the US frequency allocation plan, it illustrates how crowded the accessible frequency spectrum is.

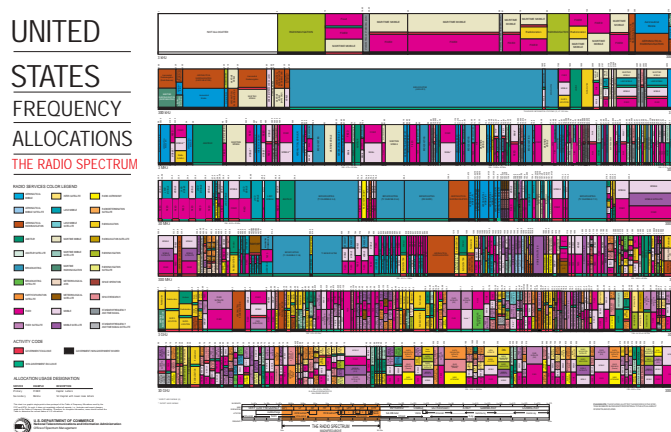


Figure 9: Frequency allocation plan US [5]

In Germany the frequency allocation plan is administered by the *Bundesnetzagentur*. Recently there were several changes, which affected many citizens. The changes originated in the need for new frequency channels in order to supply rural areas with wide-band internet connections. It is often more cost efficient to supply internet wireless for example from the closest train glass fiber network, than to rewire every village. And of course frequencies are needed to supply more people with higher data rates for their new technical gadgets .

The analog transmission of television channels was not very frequency efficient because the coding and amplitude modulation schemes were out-of-date. Therefore analog television was replaced by digital television. At the end of 2012 the analog television channel was turned of and everyone who had not done so before had to buy a digital receiver in order to decode the digital television channel. The thereby gained frequency spectrum is called *Digitale Dividende*.

Further, frequencies which were used for wireless microphones were sold by auction from 2015 in order to enhance the broadband internet. Therefore many theaters have to buy new equipment, which operates at the new frequencies assigned to wireless audio equipment. If a theater wants to use wireless microphones it has to buy a channel from the *Bundesnetzagentur*. The *Bundesnetzagentur* allots then a channel not already used by a close-by theater to the theater.

There are also open-access frequency channels, but as they can be used by everyone, the signal can be distorted by other users using the same channel.

The frequency allocation gets even more complicated as multinational agreements are required. Noone wants to throw away his mobile phones, radios, WLAN cards if he crosses a border just because the countries have different frequency allocation plans. And it also shouldn't be possible to disturb for example the Hungarian police radio with an American wireless microphone.

1.2 The analog-front end

In the last section, it was explained how information can be coded on an electromagnetic wave. In this section a closer look is taken to the analog-front end, which is the physical implementation of the digital coding. The analog front-end consists of the transmitter and the receiver. In the beginning of this section the main microwave components are described. Then an example for a traditional receiver and transmitter architecture is given. Finally a modern transmitter architecture is described.

1.2.1 Microwave components

Figure 10 shows common microwave components used for transmitter and receiver architectures and their symbolic representations.

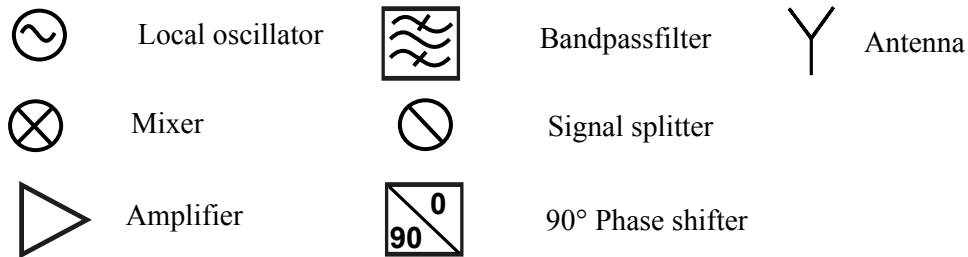


Figure 10: Symbols for common microwave components

The *local oscillator* generates a sinusoidal wave of a certain frequency and amplitude. In some applications the oscillator frequency can be changed. The most important property of the local oscillator is the frequency stability.

The ideal *mixer* multiplies two input signals with each other.

The ideal *amplifier* increases the signal power and multiplies the input signal by a constant factor larger than one. The most important properties of the amplifier are the noise figure and the linearity.

The ideal *bandpassfilter* blocks all frequency components of the signal which are lying outside of the passband.

The ideal *3-dB signal splitter* splits the input signal in two signals which each have half the power of the input signal.

The ideal *90° phase shifter* splits the input signal in two signals with half the signal power like the 3-dB signal splitter. Further the phase of one of the output signals is shifted by 90°. If for example the input of the phase shifter is a sine with amplitude 1, the outputs are a cosine and a sine each with amplitude $\frac{1}{2}$.

The *antenna* can receive free-space electromagnetic waves and transform them into guided electromagnetic waves or it can emit guided waves to the air.

There are two main technologies to guide electromagnetic waves, so-called waveguides and microstrips.

The antenna, the microwave bandpass filter, the signal splitter and the phase shifter are passive components, because they contain neither ferrites nor energy supplies. Therefore they operate in both directions. This means that a transmitting antenna is always also a receiving antenna.

1.2.2 Traditional receiver and transmitter architecture

Figure 11 and 12 illustrate an example for a traditional receiver and transmitter design. The presented design is called a super heterodyne single conversion architecture. The first observation is that the receiver and the transmitter have a similar architecture, the main difference between transmitter and receiver is the reversed signal flow direction and the additional amplifier in the transmitter block diagram, that is located right in front of the antenna. This amplifier is called an RF power amplifier.

In the receiver block diagram, the signal frequency spectrum is plotted at the in blue indicated test points. The first picture shows the channel frequency band. The second picture shows the signal after it was multiplied with the local oscillator. In the frequency spectrum the multiplication with a sine corresponds to shifting the frequency spectrum to the left and the right by the local oscillator frequency.

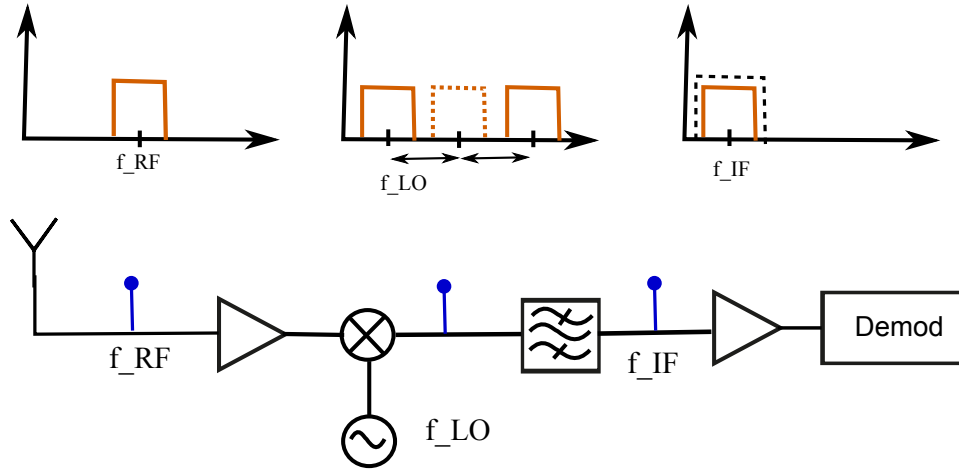


Figure 11: Block diagram: Receiver

The desired channel is selected with the help of the bandpass filter. Another channel can be selected by adjusting the local oscillator frequency.

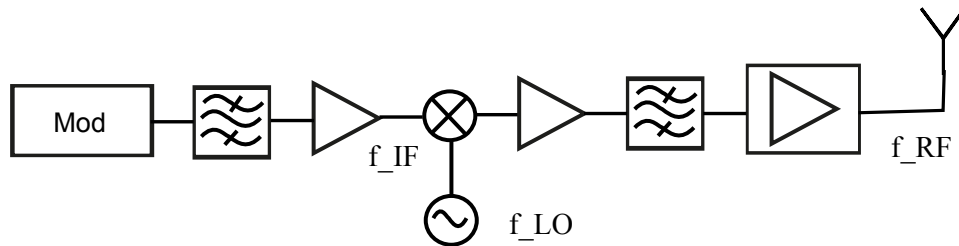


Figure 12: Block diagram: Transmitter

1.2.3 Modern transmitter architecture

In recent years, the digital progress, especially faster switching speeds, made it possible to integrate large parts of the transmitter architecture into the digital domain. The advantages of the digital integration are smaller components, lower production costs and the possibility to use more complex signal processing algorithms as for example a digital predistortion. The current trend in microwave engineering is to integrate all components into a single integrated circuit [28].

Figure 13 shows an example for such a modern transmitter architecture. A transmission path and a measurement path can be distinguished. The measurement path is required for the digital predistortion identification. First the transmission path is discussed. The transmitter receives information, in the first block, the baseband signal processing block, the information is mapped to a series of complex numbers. The mapping includes the coding, modulation and pulse-shape filtering of the signal. The complex numbers are represented by their real and an imaginary part. Then the signal is predistorted and converted to the analog domain. The real and imaginary parts are realized by mixing the signals with 90° phase shifted sinusoidal waves. Finally the signal is bandpass filtered, amplified and transmitted. The measurement path, measures the signal after the radio frequency power amplifier. The measured signal is attenuated, bandpass filtered, splitted, downconverted and converted to the digital domain. In the digital domain the in- and output of the digital predistortion block, is a sequence of complex numbers.

1.3 Problem description

A power amplifier has two desirable properties: linearity and efficiency. Linearity is important because a nonlinear component distorts the signal and broadens the frequency spectrum. Efficiency is equally

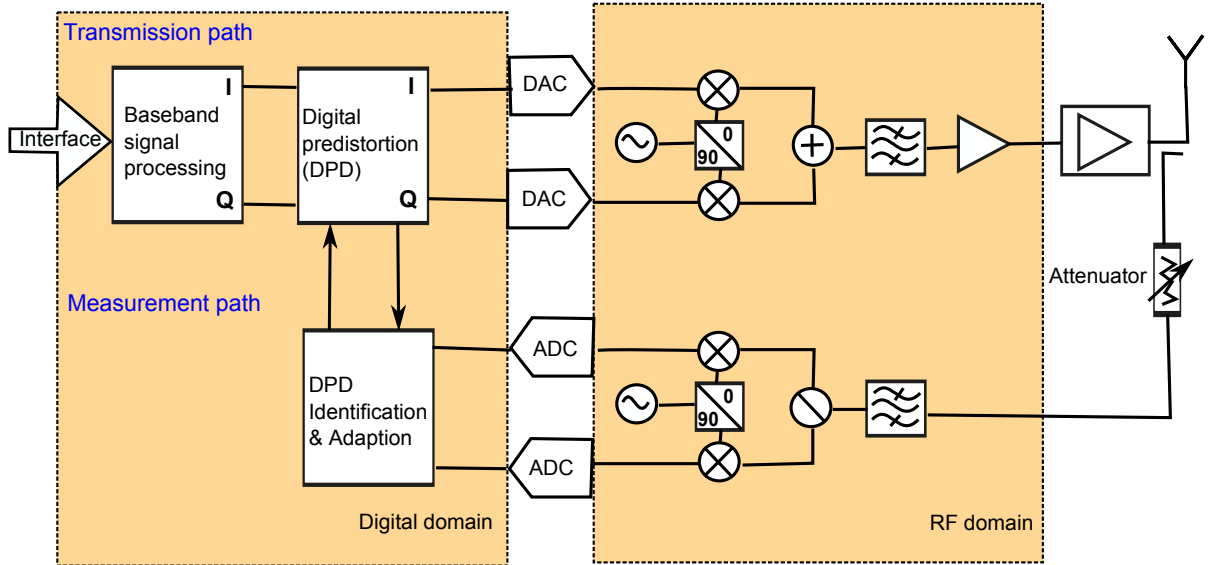


Figure 13: Block diagram: Modern transmitter architecture

important. In many wireless transmission systems power amplifiers are the most power consuming elements [22]. Mobile phone providers desire efficient power amplifiers because they operate a huge number of transmitters, 24 hours per day . Vodafone alone claims to have 33.000 base stations in Germany [26], each containing multiple amplifiers.

Unfortunately there is a trade-off between efficiency and linearity. The efficiency of the PA depends on the operation point. The linear region doesn't coincide with the efficient region. In order to use the efficient region and achieve good linearity properties the nonlinearity of the PA has to be compensated. There are several methods to compensate the nonlinearity, for example feedback control or analog predistortion. In this thesis the focus lies on a method called digital predistortion. A digital predistortion is a nonlinear component, located upstream of the power amplifier in the digital domain, which compensates the nonlinearity of the PA.

In academia the most common predistortion identification technique is called indirect learning architecture (ILA). It is well known that ILA doesn't find the optimal coefficients for the optimization problem. The suspected reasons are the cascade order problem and the noise bias problem. The focus of this master thesis lies on suggesting and testing amendments to ILA in order to overcome the above mentioned problems and to find coefficients which are closer to the optimal solution.

Scope The first suggested improvement is to augment the least-squares problem with a weighting term. Improved identification results with a weighting term were also observed in [17]. But in contrast to their paper, we are going to derive the weighting term by approximating the post-replica optimization problem by a least-squares problem. This approximation yields a weight, which is the derivative of the magnitude model. So instead of approximating the predistortion with the postdistortion, the post-replica optimization problem is approximated by a least-squares problem. Proceedingly this technique will be called *weighted indirect learning architecture (WILA)*.

The second often discussed problem with ILA is a biased estimate. Because the role of input and output signal is switched, there is input noise instead of output noise, which leads to a biased least-squares estimate. In [20] it is suggested that forward modeling, generating a noise-free data set and applying ILA mitigates these effects best. In this thesis we are going to further investigate this technique. As estimating a model introduces an additional error source, we are trying to find out for which noise level the extra modeling error is smaller than the compensated bias error. This technique will be called *model-based indirect learning architecture (MILA)*.

Finally during the simulations another observation was made. If ILA is applied iteratively and no further precautions are taken, the identified polynomial might be extrapolated. Extrapolation means that a function is evaluated outside of the area for which it was identified. This is the case if the predistortion is evaluated for values whose magnitude is larger than the largest magnitude of the identification data set. Outside of the identification area the function is not supported by any data points and may assume arbitrary values.

Limitations In this thesis only memoryfree complex PA models are investigated. Typical PA measurement data shows non-neglectable memory effects. So it seems likely that the same results will be obtained for memory-based PA models, but further investigations are necessary. For a memory-based case the basic structure stays the same, only the set of basis functions has to be extended, including also functions with input values with different timestamps.

Key results The key results of this thesis can be summarized:

- The smaller the backoff (or respectively the more of the compressed region is linearized) the more severe does the cascade order problem become. Therefore the improvements with WILA are larger for a smaller backoff. In the simulations they ranged up to 23 dB for the lowest tested backoff and were insignificant if the power amplifier function was already close to linear.
- For a noise-free DPD identification we can obtain an NMSE of -70 to -50 dB. To exploit this potential with a model-based technique for example MILA, the model identification error has to be in the same order of magnitude independent of the noise level. Often this can only be archived by using a larger model data set than predistortion identification data set.
- By adding additional data points to the initial ILA identification data set, polynomial extrapolation can be avoided. As the extrapolation effect is typically mitigated through multiple iterations, the final result is not changed significantly. Adding an additional data set to the initial guess only increases the repeatability and security, because random extrapolation effects are avoided. However for the model identification data set additional data points are necessary to avoid extrapolation.

2 Amplifier modeling

The RF power amplifier is the main source of nonlinearity in the transmitter architecture. In order to simulate and test digital predistortion identification algorithms, amplifier models are required. Therefore in this section amplifier models are described, at first the typical amplifier characteristics are discussed, then parameter sets for some common parametric memoryfree and memorybased models are given, then basis function amplifier models are described. Thereafter it is explained how the amplifier model coefficients can be identified. And finally performance metrics to evaluate the linearity of the power amplifier are presented.

2.1 Amplifier characteristics

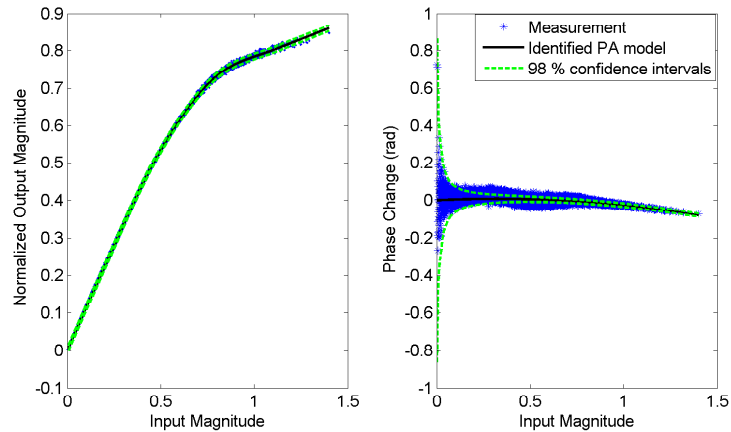


Figure 14: PA measurement data and confidence intervals. The data is normalized and was identified with a multitone signal with a relative bandwidth of 0.1.

A typical way to display a memoryless power amplifier function is a so-called amplitude-to-magnitude (AM/AM) and phase-to-magnitude (AM/PM) plot, see Figure 14 and 15. In this plots the output magnitude and the phase change are plotted versus the input magnitude. For a memoryless power amplifier function

$$y(t) = A(|x(t)|)e^{i(\Phi(|x(t)|)+\angle x(t))}, \quad (1)$$

$A(|x(t)|)$ denotes the output magnitude and $\Phi(|x(t)|)$ denotes the phase change. It is assumed that output magnitude and phase change depend only on the input magnitude and not on the phase of the input signal. This is true since a phase shift at the input of the amplifier leads to the same phase shift at the output, due to the time-invariance of the amplifier.

Typically a power amplifier is linear for small input magnitude values, while for larger input magnitude values the signal is compressed as shown in Figure 14. In many data sheets the 1-dB compression point is given in order to quantify the linear region of the power amplifier. The 1-dB compression point describes either the input or output magnitude for which the output signal power is 1 dB smaller than the desired linearly amplified output signal power.

2.2 Amplifier models

Amplifier models can be divided into memoryfree and memorybased models. Furthermore they can be distinguished by their complexity. Parametric models have only few parameters and can often only describe a specific power amplifier type. On the other hand linear basis function models are more complex. But they can be fitted to any type of power amplifier by selecting suitable basis functions. In the following section, first common parametric memoryless power amplifier models, then parametric memorybased power amplifier models are presented. Finally the structure of a linear basis function model is described and common basis functions for power amplifier modeling are discussed.

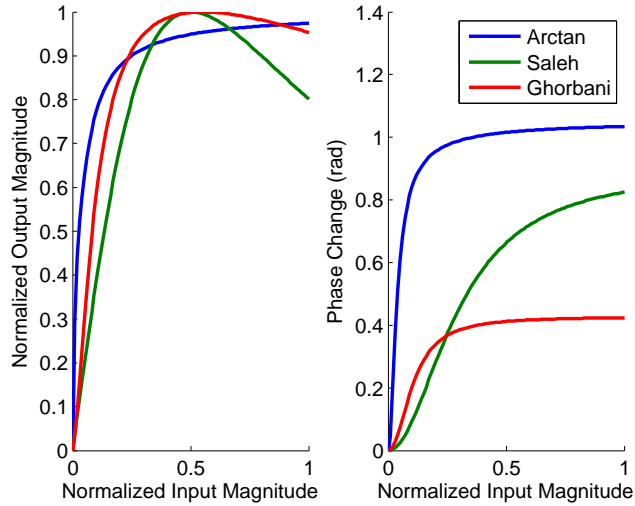


Figure 15: Normalized AM/AM (amplitude-to-magnitude) and AM/PM (phase-to-magnitude) plots for memoryless Arctan, Saleh and Ghorbani power amplifier model.

Table 1: Arctan model parameters [23]

γ_1	γ_2	φ_1	φ_2
$8.00335 - j4.61157$	$-3.77167 + j12.037$	2.26895	0.8234

2.2.1 Parametric memoryless power amplifier models

During the master thesis several papers were analyzed, proceedingly a collection of memoryless power amplifier models used for simulation studies is presented. Subsequently the equations and parameters of the Arctan, the Saleh and the Ghorbani model are presented. Figure 15 shows the AM/AM and AM/PM plot for those models.

Arctan PA model The output of the ARCTAN model is given by

$$y(t) = \left(\frac{\gamma_1 \arctan(\varphi_1 |x(t)|) + \gamma_2 \arctan(\varphi_2 |x(t)|)}{|x(t)|} \right) x(t), \quad (2)$$

the parameters are fitted to a PM2105 RFIC PA and are given in Table 1 [23].

Saleh PA model The Saleh model [25] is designed to describe a traveling wave tube amplifier (TWT),

$$y(t) = \frac{x}{|x|} A(|x(t)|) e^{i\Phi(|x(t)|)} \quad (3)$$

$$A(|x(t)|) = \frac{\alpha_a |x(t)|}{1 + \beta_a |x(t)|^2} \quad (4)$$

$$\Phi(|x(t)|) = \frac{\alpha_\Phi |x(t)|^2}{1 + \beta_\Phi |x(t)|^2}. \quad (5)$$

The parameters identified by Saleh are given in Table 2.

Table 2: Saleh model parameters [25]

α_a	β_a	α_Φ	β_Φ
1.9638	0.9945	2.5293	2.8168
1.6623	0.0552	0.1533	0.3456
2.1587	1.1517	4.0033	9.1040

Table 3: Ghorbani model parameters [19]

α_a	β_a	γ_a	δ_a	α_Φ	β_Φ	γ_Φ	δ_Φ
8.1081	6.5202	1.5413	-0.0718	4.6645	10.88	2.0965	-0.003

Ghorbani PA model The Ghorbani model is similar to the Saleh model, but has more degrees of freedom, and is claimed to be more suitable for Solid State Power Amplifier (SSPA) [19].

$$y(t) = \frac{x}{|x|} A(|x(t)|) e^{i\Phi(|x(t)|)} \quad (6)$$

$$A(|x(t)|) = \frac{\alpha_a |x(t)|^{\gamma_a}}{1 + \beta_a |x(t)|^{\gamma_a}} + \delta_a |x(t)| \quad (7)$$

$$\Phi(|x(t)|) = \frac{\alpha_\Phi |x(t)|^{\gamma_\Phi}}{1 + \beta_\Phi |x(t)|^{\gamma_\Phi}} + \delta_\Phi |x(t)|. \quad (8)$$

Parameters for a GaAs FET SSPA are given in Table 3.

2.2.2 Parametric memorybased power amplifier models

For many power amplifier types, measurement data shows considerable memory effects. Figure 14 shows a measured data set and the corresponding 98 % confidence intervals. Especially for the AM/PM plot, the variance of the data is larger than the variance caused by the measurement noise. This indicates neglected memory effects. How much the memory effects are visible in the AM/AM and AM/PM plots depends on the bandwidth of the input signal. For a larger bandwidth the memory effects are better visible. This can be explained as follows, for a continuous function memory effects mean that the function is not only dependent on x but also on derivatives of x , a bandpassfilter limits the magnitude of those derivatives and therefore the memory effects are less visible. The main sources for memory effects are currents in the bias network and thermal effects. Thermal effects can originate for example from changes in average input power. The time constants associated with thermal effects are usually larger than those associated with network effects [9]. Typically thermal effects are not modeled. Instead the PA model and the DPD coefficients are identified under normal operating conditions or adaptive techniques are applied to adapt the model to varying operating conditions.

Many parametric memorybased power amplifier models have a Wiener (W) or Wiener-Hammerstein (WH) structure. A Wiener structure consists of a linear dynamical system followed by a static nonlinearity. For a Wiener-Hammerstein structure the static nonlinearity is located between two linear dynamical systems. In most of the investigated papers, the static nonlinearity is selected from one of the memoryless PA models, described in the last section. The linear dynamical system is described by a finite impulse response (FIR) or infinite impulse response (IIR) filter. An overview over parametric memorybased power amplifier (PA) models is given in Table 4.

2.2.3 Linear basis function model

Linear basis function models play an important role for power amplifier modeling as well as predistortion modeling, because they can be fitted to any type of function by selecting an appropriate set of basis functions and model order. In this paragraph, first the structure of a linear basis function model is given and then the most common set of basis functions power amplifier models is presented.

A linear basis function model is a linear combination of a set of basis functions

$$f(\mathbf{x}, a) = \sum_{k=1}^N a_k \Phi_k(\mathbf{x}). \quad (9)$$

Table 4: Parametric memorybased PA models

Paper	Structure	Nonlinearity	Memory type	Memory order	Filter coefficients
[23]	W-H	Arctan	IIR	(2,1),(1,1)	See paper eq. (31)
[18]	W	Saleh	FIR	2	See [16] eq. (13)
[8]	W	Saleh	FIR	0 – 8	not given

Legend

Structure: Wiener(W), Wiener-Hammerstein (W-H)

Memory order: FIR filters-memory length.

IIR filters- degree of nominator N and denominator D (N,D)

where N is the model order, a_k are the model coefficients, Φ_k are the basis functions and \mathbf{x} is the input signal. If the basis functions are memorybased \mathbf{x} is a vector containing the last $M + 1$ samples, where M is the memory order. While the model is linear in its parameters a_k , the basis functions $\Phi_k(x)$ are nonlinear functions of x , and a large set of amplifiers can be represented by this type of model.

Power amplifier (PA) model For power amplifiers modeling a common linear basis function model is the memory polynomial model [21] [12] ,

$$y(n) = \sum_{m=0}^{M-1} \sum_{k=1}^K a_{mk} x(n-m) |x(n-m)|^k. \quad (10)$$

In the same notation as before the memory polynomial basis functions are given by

$$\Phi_{MP,mk}(\mathbf{x}) = x(n-m) |x(n-m)|^k, \quad (11)$$

and for a memoryless power amplifier the polynomial basis functions are given by

$$\Phi_k(\mathbf{x}) = x(n) |x(n)|^k. \quad (12)$$

The linear basis function PA model is given by

$$\hat{f}_{PA}(\mathbf{x}, a) = \sum_{k=1}^{N_{PA}} a_k \Phi_k(\mathbf{x}). \quad (13)$$

Beside the selection of the basis functions the model order influences the quality of the linear basis function model; if the selected order is too high the computational effort for the identification increases and there is a chance that the model is overfitted to noise, while if the model order is too low the model can't represent the function accurately enough. In the investigated papers, the memory order was chosen between $M=2$ and $M=5$ and the polynomial order was chosen between $K=5$ and $K=9$, see Table 5.

Table 5: Memory order and polynomial order for memory polynomial models

Paper	Memory order M	Polynomial order K
[14]	M=2	K=5
[23]	M=2	K=7
[18]	M=3	K=5
[7]	M=5	K=9

A more general set of basis functions, allowing also interactions between samples with different timesamps, is the Volterra series. However it is impractical because of the high number of basis functions and the resulting high computational effort. A Volterra series based model with a reduced number of basis function is given in [31].

2.3 Model identification

In order to obtain the coefficients a_k of the PA model (13) from noisy measurement data,

$$\mathbf{y}_\eta = f_{PA}(\mathbf{u}) + \eta \quad (14)$$

the least-squares optimal solution is given by

$$\begin{aligned} Z &= [\Phi_1(\mathbf{x}), \dots, \Phi_{N_{PA}}(\mathbf{x})] \\ \hat{\mathbf{a}} &= (Z^H Z)^{-1} Z^H \mathbf{y}_\eta, \end{aligned} \quad (15)$$

where $[\cdot]^H$ denotes the Hermitian transpose, N_x is the length of the measured sequence, \mathbf{x} is the $N_x \times 1$ input data vector \mathbf{y}_η is the $N_x \times 1$ measured output signal, Φ_i are the basis functions, Z is a $N_x \times N_{DPD}$ matrix and \mathbf{a} is the parameter vector. For the noise-free case the least-squares solution minimizes

$$\mathbf{a}^* = \arg \min_{\mathbf{a} \in \mathbb{C}^{N_{PA}}} \sum_{i=1}^{N_x} |f_{PA}(u[n]) - \hat{f}_{PA}(u, \mathbf{a})|^2. \quad (16)$$

It can be shown that the estimate is unbiased, which means that for the noisy case the expectation value (mean value if experiment is repeated an infinite amount of times) of $\hat{\mathbf{a}}$ is equal to the optimal coefficients identified for the noise-free case

$$E[\hat{\mathbf{a}}] = \mathbf{a}^*. \quad (17)$$

2.4 Performance metrics

The main error source of power amplifiers is their nonlinearity. The two performance metrics used within this thesis to measure the nonlinearity of the power amplifier are the normalized mean square error (NMSE) and the adjacent channel power ratio (ACPR). A nonlinear component broadens the frequency spectrum and therefore can distort the signal in the adjacent channel. The ACPR is a measure for the out-of-channel distortion. The signal itself is also distorted by the nonlinearity, the NMSE is a measure for the in-band distortion. The NMSE is the mean squared-error between desired output signal and the output signal

$$e[n] = y[n] - K_\gamma x[n], \quad (18)$$

where K_γ is the desired linear gain, divided by the mean output power,

$$NMSE_{dB} = 10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} |e[n]|^2}{\sum_{n=0}^{N-1} |y[n]|^2} \right). \quad (19)$$

The ACPR is defined as the adjacent channel power divided by the reference channel power, see Figure 16,

$$ACPR_{dB} = 10 \log_{10} \left(\frac{P_{adj}}{P_{ref}} \right). \quad (20)$$

The figure illustrates that between two channels a channel gap can be defined. The width of the channel gap differs for different standards. The ACPR values in the result section were obtained without a channel gap.

Because the error depends on the linearisation area and on the input signal distribution it is often plotted against the output backoff (OBO). The output backoff is defined as the maximum possible signal power divided by the output power

$$OBO_{dB} = 10 \log_{10} \left(\frac{P_{max}}{P_{out}} \right) = 10 \log_{10} \left(N \frac{|y_{max}|^2}{\sum_{n=0}^{N-1} |y|^2} \right). \quad (21)$$

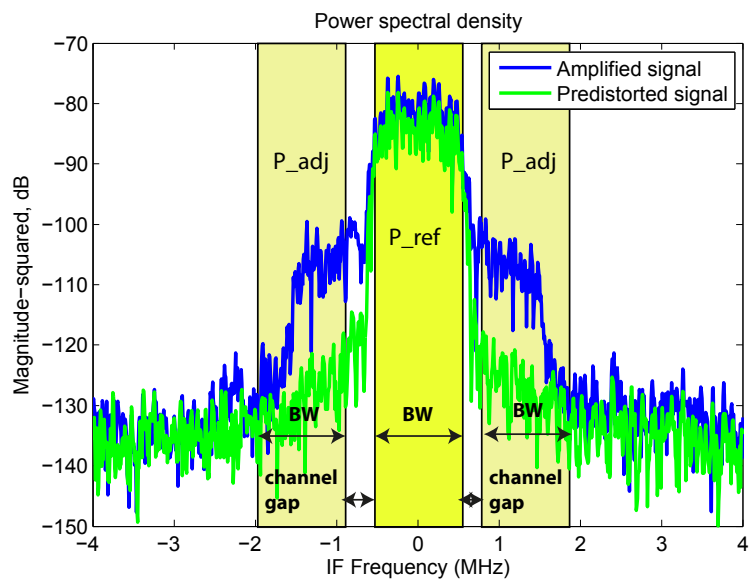


Figure 16: Adjacent channel power (ACPR)

3 Digital predistortion (DPD)

3.1 The general idea of DPD

Many problems are associated with nonlinear power amplifier behavior. In order to compensate the non-linearity of the power amplifier the signal can be predistorted with a digital predistortion, as illustrated in Figure 17.

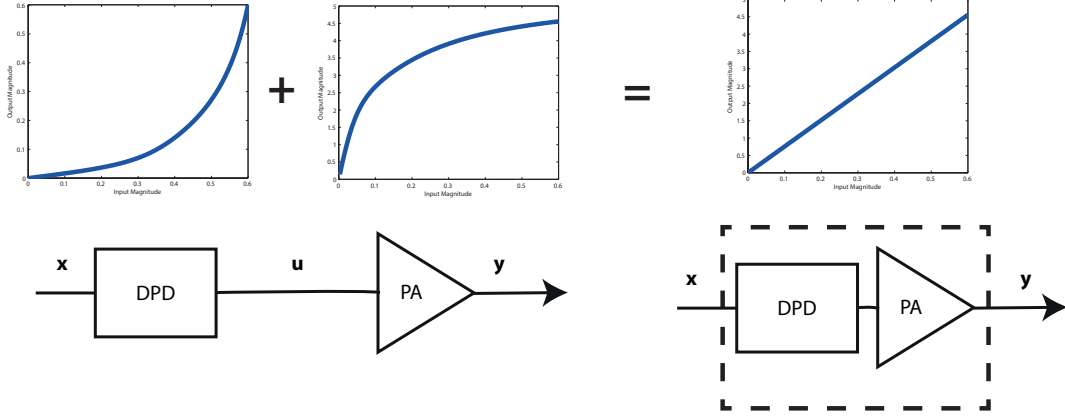


Figure 17: Block diagram: Digital predistortion

Colloquially the digital predistortion identification problem can be formulated as follows: for a nonlinear power amplifier function the corresponding nonlinear predistortion function is searched such that the composition of predistortion and power amplifier function yields a linear function.

3.2 Mathematical formulation of the DPD identification problem

In order to describe the predistortion identification problem mathematically precise, the following four points have to be defined: the predistortion subspace, the optimization criteria, the linear gain and the maximum input magnitude (linearization area).

1. Defining a predistortion model subspace.

There are different types of predistortions for example look-up-tables or linear basis function models. In order to formulate the identification problem, the function class in which the optimal predistortion is searched needs to be defined.

In this thesis the digital predistortion is given by a linear basis function model

$$f_{DPD}(\mathbf{x}, \theta) = \sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(\mathbf{x}), \quad (22)$$

where N_{DPD} is the model order, θ_k are the model coefficients, Ψ_k are the basis functions and \mathbf{x} is the input signal. H_{DPD} describes the model subspace or the space of all possible functions for a given set of basis functions. Many papers focus on the optimal choice of basis functions. In this thesis it is assumed that the model order and the basis functions are fixed and the goal is to find the optimal predistortion coefficients. For the simulations memory-polynomial basis functions were selected.

2. Defining the optimization criteria.

As the model subspace does not contain all existing monotone functions, it is clear, that the goal cannot be to find the ideal predistortion, in the sense that the combination of predistortion and amplifier yields a linear function. The ideal predistortion may not be contained in the model

subspace. Therefore an optimization problem needs to be formulated. It should have the attribute that if the ideal predistortion is contained in the model subspace, it is the unique solution of the optimization problem. Herein the following discrete optimization criteria is used

$$J = \frac{1}{N} \sum_{n=1}^N |f_{PA}(f_{DPD}(x[n])) - K_{\gamma}x[n]|^2, \quad (23)$$

where N is the number of measurement samples, K_{γ} is the desired linear gain, $x[n]$ are the complex input samples and $y[n] = f_{PA}(f_{DPD}(x[n]))$ are the complex output samples, as illustrated in Figure 17. This means, the goal is to minimize the mean-squared error between linearly amplified signal and predistorted signal.

The continuous formulation is the limit of the discontinuous formulation if the number of samples is increased to infinity $N \rightarrow \infty$, it is given by

$$\min_{f_{DPD} \in H_{DPD}} \int_{I_x} p_x(x) (f_{PA}(f_{DPD}(x)) - K_{\gamma}x)^2 dx, \quad (24)$$

where $p_x(x)$ is the probability density function of the input data, K_{γ} is the desired linear gain and I_x is the linearization area containing all possible input values and H_{DPD} is the DPD subspace. For simulation purposes, the input probability density function can be approximated by complex Gaussian noise, as described in paragraph 1.1.4. In order to simplify the calculation, it is assumed that the linearization area is a circle in the complex plane whose center is the origin.

3. Choosing the desired linear gain.

The desired output signal is a linear function of the input signal

$$y_{des}(x) = K_{\gamma}x. \quad (25)$$

There are two reasonable ways to choose the desired linear gain K_{γ} . The first one is to choose K_{γ} , such that the predistortion doesn't modify the average output power. The second one is to choose K_{γ} , such that the predistortion doesn't modify the maximum output magnitude. Both ways are equally good, but require different routines to calculate the desired linear gain and find an initial guess for the predistortion. For the first choice, the output signal distribution looks similar with and without DPD. This is important for the convergence of the initial guess. The same property can also be archived by reducing the input power for the initial guess. This means instead of $f_{DPD,init}(x) = x$, the initial predistortion is $f_{DPD,init}(x) = K_{init}x$. The second choice does not modify the input range seen by the power amplifier.

4. Defining the linearization area.

The linearization area or maximum input magnitude is given by the implementation of the modulation scheme and the maximum number of users. This upper bound can be reduced with a crest factor reduction scheme. For simulation purposes the upper bound can be implemented by clipping the input signal at a certain magnitude level.

The predistortion identification problem can be summarized as follows: Given a desired linear gain K_{γ} , a linearization area I_x and a nonlinear, complex power amplifier model

$$f_{PA}(u) : I_u = C(0, b_u) \in \mathbb{C} \rightarrow I_y = C(0, b_y) \in \mathbb{C}, \quad (26)$$

the goal is to find the coefficients θ_k of the digital predistortion linear basis function model (22)

$$\begin{aligned} f_{DPD}(x) : I_x = C(0, b_x) \in \mathbb{C} &\rightarrow I_u = C(0, b_u) \in \mathbb{C}, \\ f_{DPD}(x) &:= \sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(x) \end{aligned} \quad (27)$$

such that the weighted-mean-squared error between desired output and predistorted output is minimized

$$\begin{aligned}\theta^* &= \arg \left(\min_{\theta \in \mathbb{C}_{DPD}^N} \lim_{N \rightarrow \infty} \sum_{n=1}^N |f_{PA}(f_{DPD}(x[n])) - K_\gamma x[n]|^2 \right) \\ &= \arg \left(\min_{\theta \in \mathbb{C}_{DPD}^N} \int_{I_x} p_x(x) |f_{PA}(f_{DPD}(x)) - K_\gamma x|^2 dx \right).\end{aligned}\quad (28)$$

3.3 Digital predistortion identification with the indirect learning architecture (ILA)

Indirect learning architecture is one of the standard methods used in academia to identify the coefficients of a linear basis function predistortion model, it is for example used in [21], [15], [12], [10] and [30].

3.3.1 Predistortion versus postdistortion

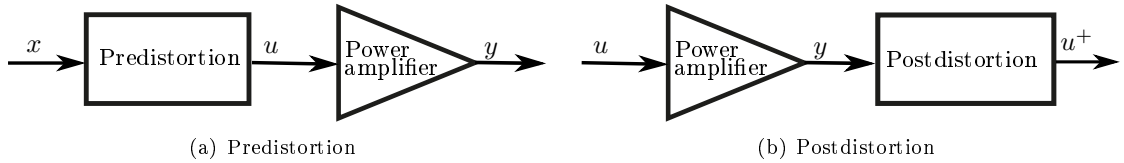


Figure 18: Block diagram predistortion and postdistortion

In order to motivate indirect learning architecture (ILA), the pre- and postdistortion concepts are contrasted. Figure 18 shows a block diagram of a post- and of a predistortion. The difference between pre- and postdistortion is the component order. The predistortion distorts the signal before the amplifier, whereas the postdistortion distorts the signal after the amplifier. According to equation (23) the predistortion optimization problem is

$$\min_{\theta \in \mathbb{C}^{N_{DPD}}} \frac{1}{N} \sum_{n=1}^N \left| f_{PA} \left(\sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(x[n]) \right) - K_\gamma x[n] \right|^2. \quad (29)$$

The equivalent postdistortion optimization problem, obtained by taking the mean-squared error between postdistorted output and desired output, is

$$\min_{\theta \in \mathbb{C}^{N_{DPD}}} \frac{1}{N} \sum_{n=1}^N \left| \sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(f_{PA}(u[n])) - K_\gamma u[n] \right|^2 \quad (30)$$

The postdistortion identification problem is a least-squares problem. This can be seen by reformulating the identification problem in matrix notation

$$\min_{\theta \in \mathbb{C}^{N_{DPD}}} \frac{1}{N} (\mathbf{M}(\mathbf{y})\boldsymbol{\theta} - \mathbf{u})^T (\mathbf{M}(\mathbf{y})\boldsymbol{\theta} - \mathbf{u}), \quad (31)$$

where

$$\begin{aligned}\mathbf{M}(\mathbf{y}) &= [\Psi_1(\mathbf{y}), \dots, \Psi_{N_{DPD}}(\mathbf{y})] \\ \boldsymbol{\theta} &= [\theta_1, \dots, \theta_{N_{DPD}}]^T \\ \mathbf{u} &= [K_\gamma u[1], \dots, K_\gamma u[N]]^T \\ \mathbf{y} &= [y[1], \dots, y[N]]^T \\ y[n] &= f_{PA}(u[n]).\end{aligned}\quad (32)$$

While the postdistortion problem is a least-squares problem, the predistortion optimization problem is a nonlinear optimization problem, because the coefficients θ_k are located inside a nonlinear function

f_{PA} . Compared to a nonlinear optimization problem a least-squares problem has many advantages. Firstly least-squares problems are convex, therefore the problem has no local minima or maxima. This means that a local optimum is always also the global optimum of the optimization problem. Figure 19 illustrates the difference between a convex and a non-convex optimization criteria for a one-dimensional problem. Secondly solving a least-squares problem is computationally much more efficient than solving a nonlinear optimization problem. A common way to solve nonlinear optimization problems is to approximate them iteratively by least-squares problems. Unfortunately it is not possible to implement a postdistortion. Figure 13 illustrates why. While the predistortion can be located in the digital domain,

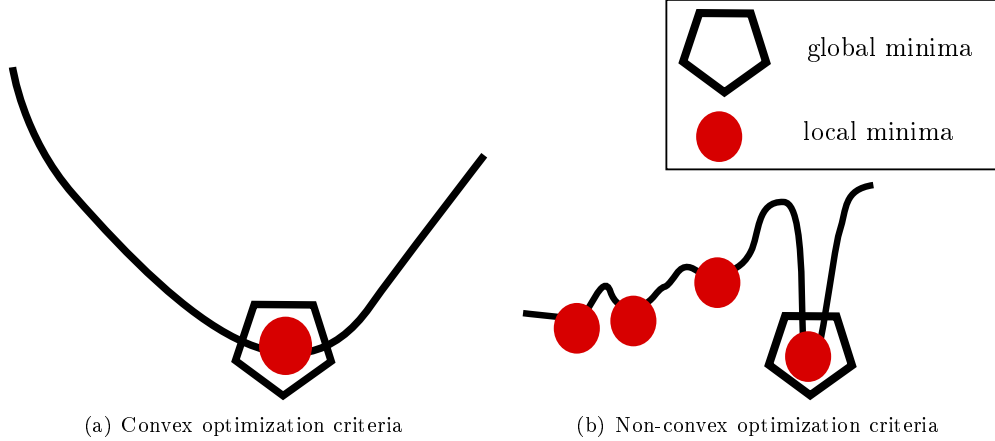


Figure 19: Comparison convex and non-convex optimization problem

the postdistortion would have to be located in the RF domain. While the signal processing infrastructure already exists in the digital domain, there is no signal processing infrastructure in the RF domain. There are several reasons why signal processing isn't done at the RF frequency. Firstly the RF frequency is higher than the normal signal processing frequency, therefore the components would need to operate at higher speeds. Secondly each application has a different RF frequency, therefore different signal processing speeds would be required for each application.

3.3.2 Indirect learning architecture

Given all the advantages the identification of a postdistortion has compared to the identification of a predistortion, it is not surprising that the underlying idea of the indirect learning architecture (ILA) is to identify the postdistortion and use it as predistortion. Before the postdistortion the signal is scaled with the reciprocal power amplifier gain, as shown in Figure 20. The interchanging of pre- and postdistortion is motivated by the fact that in the ideal case, when the postdistortion linearizes the amplifier perfectly, postdistortion and predistortion are interchangeable. Subsequently this will be shown. Assume that

$$f_{DP_{ostD}}\left(\frac{f_{PA}(u)}{K_\gamma}\right) = u, \quad (33)$$

then the postdistortion can be expressed in terms of the inverse power amplifier function

$$f_{DP_{ostD}}\left(\frac{f_{PA}(u)}{K_\gamma}\right) = u = f_{PA}^{-1}(f_{PA}(u)) = f_{PA}^{-1}\left(K_\gamma \frac{f_{PA}(u)}{K_\gamma}\right), \quad (34)$$

substituting $\frac{f_{PA}(u)}{K_\gamma}$ by x yields

$$f_{DP_{ostD}}(x) = f_{PA}^{-1}(K_\gamma x). \quad (35)$$

Therefore using the postdistortion as predistortion yields the desired linear function

$$f_{PA}(f_{DP_{ostD}}(x)) = f_{PA}(f_{PA}^{-1}(K_\gamma x)) = K_\gamma x. \quad (36)$$

Figure 21 illustrates the indirect learning architecture. For the postdistortion, the optimal solution, which minimizes the least-squares error between u and $u^+ = \frac{y_n}{K_\gamma}$, is calculated. Then the identified

postdistortion is used as predistortion. The optimal solution θ of the postdistortion least-squares problem is given by

$$\begin{aligned} \mathbf{y}_\eta &= f_{PA}(\mathbf{u}) + \eta \\ Z &= \left[\Psi_1 \left(\frac{\mathbf{y}_\eta}{K_\gamma} \right), \dots, \Psi_{N_{DPD}} \left(\frac{\mathbf{y}_\eta}{K_\gamma} \right) \right] \\ \theta &= (Z^H Z)^{-1} Z^H \mathbf{u}, \end{aligned} \quad (37)$$

where $[\cdot]^H$ denotes the Hermitian transpose, \mathbf{u} is the $N_x \times 1$ actuator variable $\mathbf{u} = [u(t_1) \dots u(t_{N_x})] = [u[1] \dots u[N_x]]$, \mathbf{y}_η is the $N_x \times 1$ measured output signal, Ψ_i are the basis functions and Z is a $N_x \times N_{DPD}$ matrix.

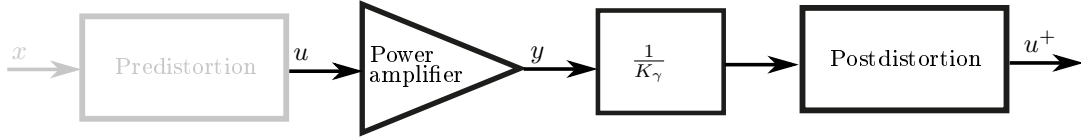


Figure 20: Postdistortion for ILA

The postdistortion identification algorithm has to be applied iteratively, because changing the predistortion modifies the signal u and hence also the optimization problem. Figure 22 shows exemplary how the probability distribution of the actuator variable u is modified by adding a predistortion. In this example the output power of the predistorted and unpredistorted signal u is kept approximately constant. Keeping the output power approximately constant is essential for the performance of the iterative algorithm. In the next paragraph it will be explained how this can be done. So starting with an initial predistortion guess, represented by a parameter vector θ^0 , the postdistortion optimization problem is solved iteratively and the predistortion is updated with the parameter vector θ^i after each iteration. Typically the initial predistortion function, represented by θ^0 , will represent a linear function

$$f_{DPD}^0(x) = \theta_0^0 x, \quad (38)$$

with a gain such that the signal power of y is approximately the signal power of the ideally predistorted output signal $\tilde{y} = K_\gamma x$. The importance of a good initial guess θ_k^0 can be illustrated with a bad example. If $f_{DPD}^0(x)$ was 0, the vector \mathbf{u} would contain only a single point and hence the problem would be singular. Note, that when discussing ILA, it will become relevant that in comparison to the standard least-squares problem

$$(\mathbf{M}(\mathbf{u})\boldsymbol{\theta} - \mathbf{y})^T (\mathbf{M}(\mathbf{u})\boldsymbol{\theta} - \mathbf{y}) \quad (39)$$

the role of input and output is interchanged for the postdistortion problem (31)

$$\min_{\boldsymbol{\theta} \in \mathbb{C}^{N_{DPD}}} \frac{1}{N} (\mathbf{M}(\mathbf{y})\boldsymbol{\theta} - \mathbf{u})^T (\mathbf{M}(\mathbf{y})\boldsymbol{\theta} - \mathbf{u}).$$

Therefore the noise acts on what is normally the noise free input of the least-squares algorithm.

3.3.3 Obtaining a good starting point for iterative ILA

As for any iterative algorithm a good starting point is essential for the convergence and performance. For ILA the output of the PA acts as input of the identification algorithm. Therefore a good starting point means that the output distribution of the PA is similar to the predistorted output distribution of the PA. This can be ensured by keeping the mean output power approximately constant during the first iteration. The output power can either be kept constant by changing the input power or by selecting the linear gain appropriately.

The proceeding for decreasing the input power for the initial guess is:

1. Measure the input power P_x

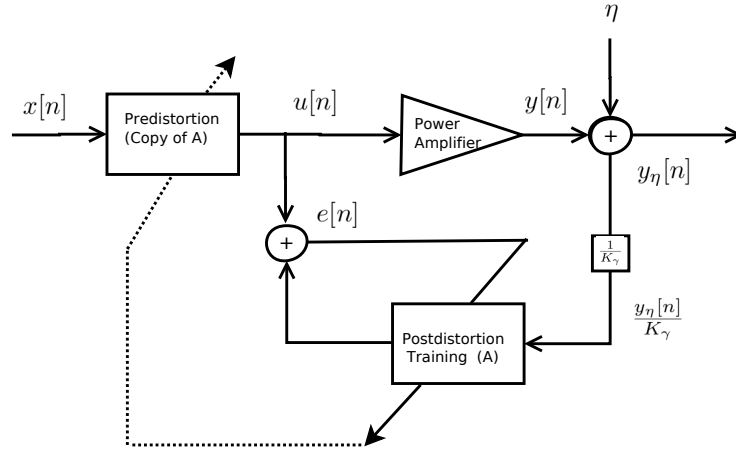


Figure 21: Block diagram: Indirect learning architecture (i.e. [13] Fig.1 or [23] Fig.6)

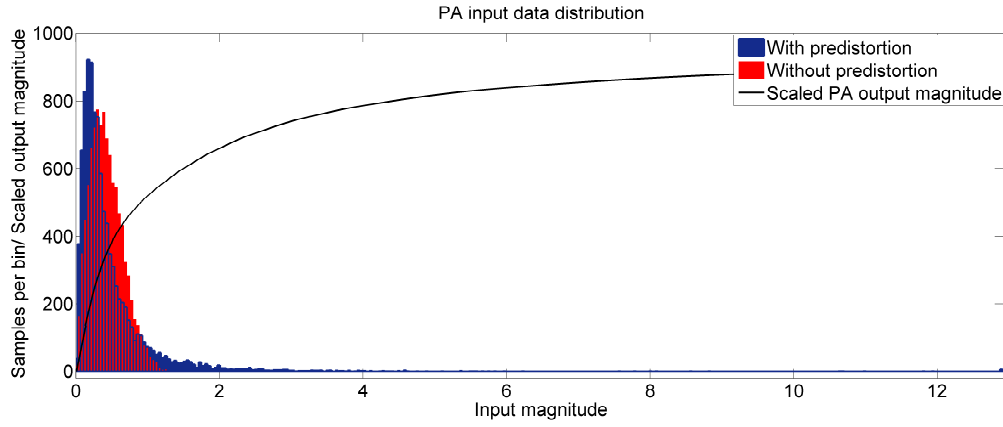


Figure 22: Histogram for the PA input magnitude $|u|$ with and without a predistortion

2. Calculate the output power for the desired linear gain

$$P_y = K_\gamma^2 P_x \quad (40)$$

3. Decrease the input power $P_{x,init}$ for the initial guess until the measured output power is approximately equivalent to the desired output power $P_{y,init} \approx P_y$

The proceeding for choosing the linear gain appropriately is:

1. Measure the input and the output power of the undistorted signal
2. Calculate the required linear gain for keeping the output power constant

$$K_\gamma = \sqrt{\frac{P_{y,mess}}{P_{x,mess}}} \quad (41)$$

3. The maximum output power of the predistorted signal is then given by $b_y = K_\gamma b_x$

This procedure is only required for the first iteration in order to find a good starting point for the iterative algorithm.

4 Discussion and analysis of problems regarding ILA

The following three problems are going to be discussed in the next paragraphs:

1. The extrapolation problem for the initial guess.
2. The cascade order problem.
3. The noise bias problem.

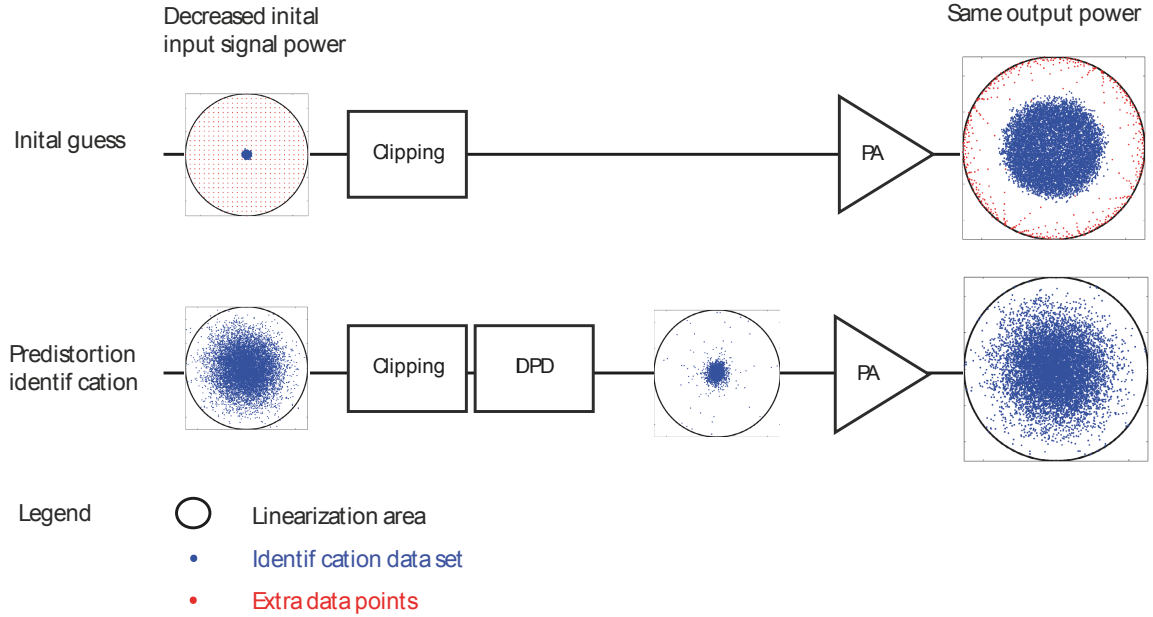


Figure 23: Input and output data distribution seen by the PA for the predistorted and the unpredistorted signal, with an decreased input power

4.1 Extrapolation problem

All three methods ILA, WILA and MILA require an initial guess of the predistortion. The reason is, that the predistortion changes the input and output data distribution seen by the power amplifier and the identification algorithm. A good starting point for the initial guess requires that the mean output power is kept approximately constant, between the first iteration without predistortion and the second iteration with the first predistortion estimate. If the mean output power is kept constant the maximum output power is changed, see Figure 23 and 24. The consequence is that the first identification data set will not cover the complete linearization area, a circle with radius $K_\gamma b_x$, there will be a ring-shaped area $K_\gamma(b_x - b_e)$, without any data points. Therefore in the identification data set there are no data points supporting the ring-shaped area $(b_x - b_e)$. The digital predistortion is only identified for the circle with radius b_e and extrapolated to the circle with radius b_x in the first iteration. How severe the extrapolation problem is depends on the backoff the smaller the backoff the larger the ring-shaped area without data points.

4.2 Cascade order problem

The cascade order problem origins from identifying the postdistortion instead of the predistortion. Also in the ideal case postdistortion and predistortion are interchangeable, they are the solutions of two different optimization problems.

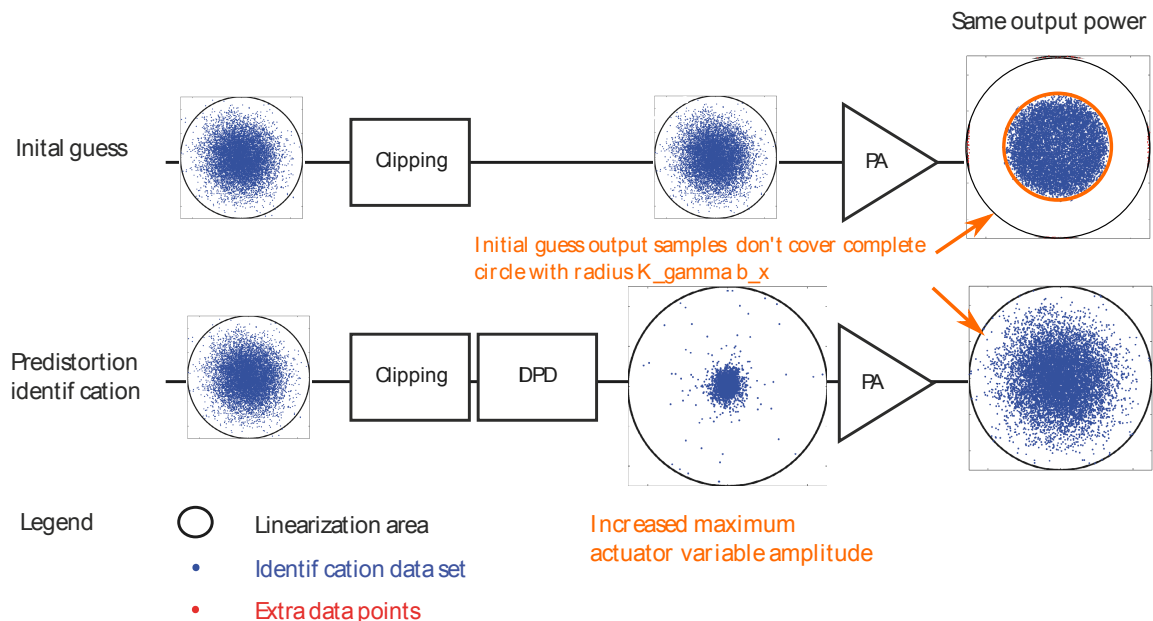


Figure 24: Input and output data distribution seen by the PA for the predistorted and the unpredistorted signal, with an adjusted linear gain

The problem is visible, when looking at the error of an ILA solution. In order to illustrate the problem, first the least-squares optimal solutions of some weighted polynomial fitting problems are shown in Figure 25. The first column gives the number of basis functions, the second column shows the function, the third column shows the weight and the fourth column shows the squared-error times the weight we^2 . Independent of the shape of the function and the weight the envelope of we^2 is approximately constant in the middle and slightly higher at the edges of the interval. A deviation in the middle of the interval will influence the left and the right side, while a deviation at the edges only influences one side, therefore the error is slightly higher at the edges.

We expect a similar behaviour for the predistortion optimization problem, namely that the envelope of the squared-error times the weight of the optimal solution is slightly higher at the edges and approximately constant in the middle. Looking at an ILA solution, see Figure 26, shows that the envelope doesn't look as expected. This indicates that the ILA solution is not the optimal solution.

4.3 Noise bias problem

The noise bias problem originates from the noise that acts on the input of the least-squares problem. A biased solution means that the expectation value of the solution and the optimal solution don't coincide. Or in other words if the number of data points is increased the solution doesn't converge to the optimal solution. Figure 27 illustrates the cause of the input noise bias: the noise averaging direction and the noise direction don't coincide. If the noise acts on the output signal, the error for the least-square estimate is

$$e = M(u)\theta - (y + \boldsymbol{\eta}) \quad (42)$$

but for ILA the role of input and output is interchanged and the noise acts on the input signal

$$e = M(y + \boldsymbol{\eta})\theta - u. \quad (43)$$

A least-square estimate with additive white output noise is unbiased. But with additive input noise, the estimate is biased, which is illustrated in Figure 28. The noise level is exaggerated to make the effect more visible. The first subplot shows the measured data set with input and output noise respectively. The top subplots show the original function (black) and the identified functions. The bottom subplots show the error between original function and identified function. For an unbiased estimate the error

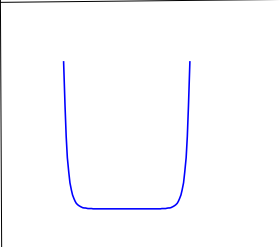
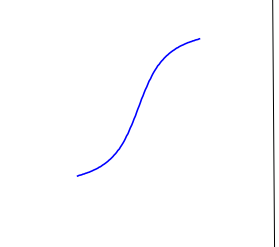
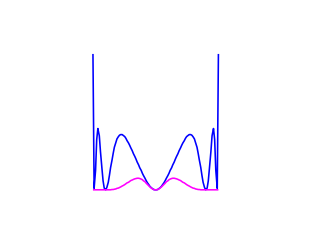
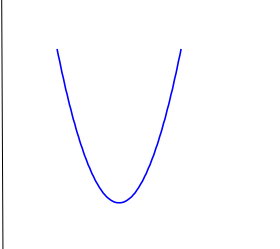
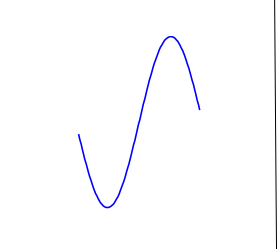
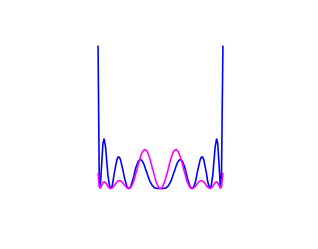
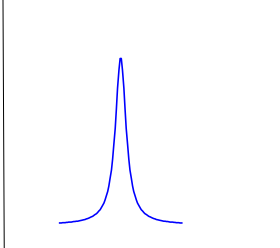
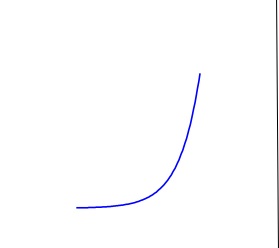
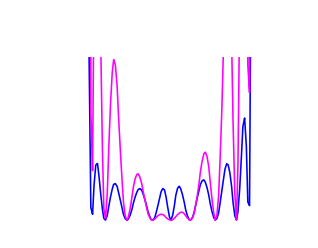
Polynomial order	Weight	Function	we^2 and e^2
5			
8			
3			

Figure 25: Weighted polynomial fitting and weighted squared-errors

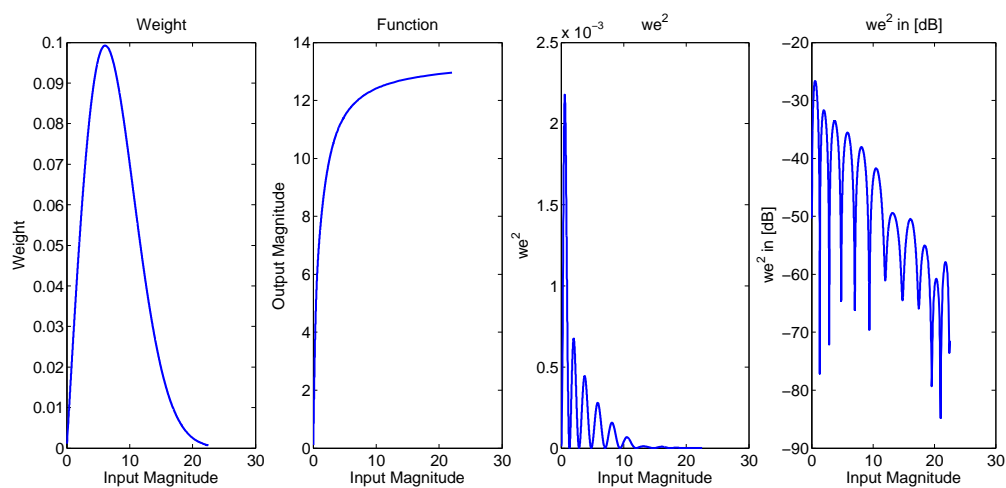


Figure 26: Weight, Function and we^2 for an ILA solution

gets arbitrary small if the number of data points is increased, but for the biased estimate the error is persistent.

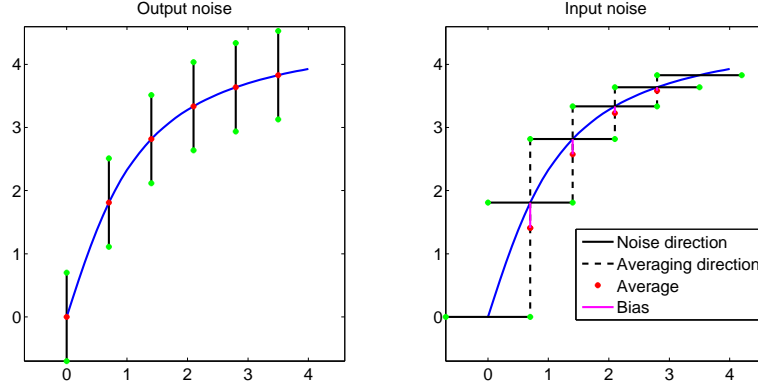


Figure 27: Input noise bias

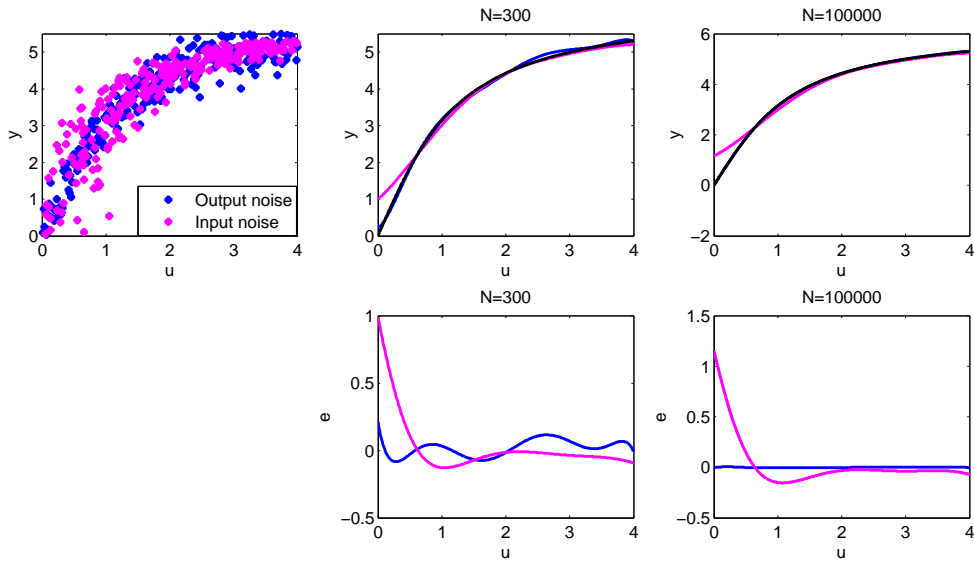


Figure 28: Bias due to additive input noise

5 Proposed solutions

5.1 Reinforcement data grid

The proposed solution for the extrapolation problem is to add an additional reinforcement data grid to the identification data set for the first iteration. How the reinforcement grid is constructed depends on whether the gain adjustment or the input power adjustment is used for the initial guess. If the input power adjustment is used the reinforcement grid can be constructed as follows:

1. After the input power was adjusted, identify the ring-shaped area that contains the least- p_L percent of data samples.
2. Enforce this area with a sample grid. The density of the grid ρ_{data} per unit area is calculated such that the number of extra data points is approximately N_{extra} .

The parameter p_L and N_{extra} can be specified in the simulation parameter script. An example for the reinforced identification data set is shown in Figure 23 for the initial predistortion guess.

If the gain adjustment is used the reinforcement grid can be constructed as follows:

1. Identify the input magnitude required to reach the output magnitude $K_\gamma b_x$

2. Enforce the ring-shaped area between maximum input magnitude and required input magnitude to reach the output magnitude $K_\gamma b_x$ with a sample grid.

Further a reinforcement data grid is necessary for the model identification, because the data distribution is very scarce at the border of the circle with radius b_u , see Figure 23 (Input to the PA with initial predistortion guess). An example that shows the importance of the reinforcement data grid for the model identification, is given in Figure 29. Without the reinforcement data grid the identified polynomial model oscillates at the border of the linearization area, with the reinforcement data grid there is no oscillation.

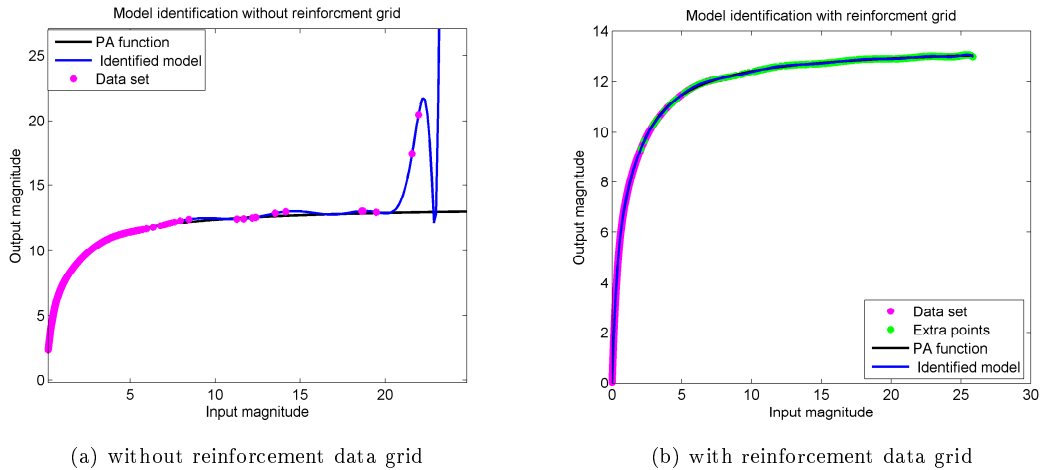


Figure 29: Extrapolation problem without border area reinforcement for the model identification

5.2 Weighted indirect learning architecture (WILA)

The big advantage of ILA over other predistortion identification techniques is the comparable small computationally complexity and the simplicity of the solution algorithm. In this section, we propose a method that will be called *weighted indirect learning architecture* (WILA). It is a way to keep this advantage, while getting rid of the cascade order problem. Figure 30 illustrates the idea behind WILA. It is similar to the one of ILA. For ILA the circuit is augmented by a copy of the DPD and then the postdistortion optimization problem is solved instead of the predistortion optimization problem. The post-distortion problem is a least-squares problem. Compared to a nonlinear optimization problem a least-squares problem has many advantages. To begin with the calculation of the optimal solution is computational much less expensive. Further a least-squares problem is a convex optimization problem. This means there are no local minima. For WILA the circuit is augmented by a copy of the DPD and the PA. The postreplica optimization problem is still a nonlinear problem, but it can be approximated by least-squares problem. In the derivation section it will be shown how this is done with the help of the Taylor series and by neglecting higher than second-order error terms.

Unfortunately this thesis only contains experimental evidence and no proof that the solution of the postreplica optimization problem is closer to the optimal solution than the solution of the postdistortion problem. Further the derivation is only given for real memory-free power amplifier functions.

5.2.1 Algorithm

The WILA algorithm is similar to the ILA algorithm with the amendment that a magnitude model of the power amplifier is identified and a weighted-least square problem is solved, the weight is the

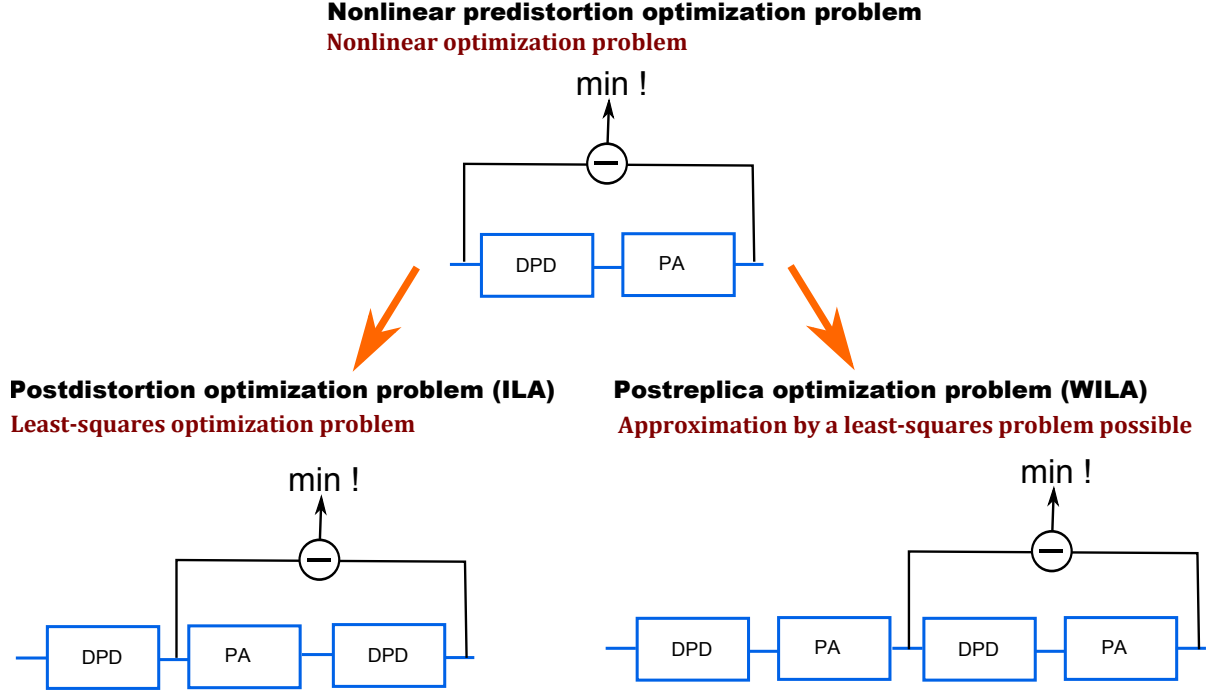


Figure 30: Comparison predistortion optimization problem, postdistortion optimization problem (ILA) and postreplica optimization problem (WILA)

derivative of the magnitude model,

$$\begin{aligned}
 \mathbf{u} &= f_{DPD}(\mathbf{x}, \theta_{init}) \\
 \mathbf{y}_\eta &= f_{PA}(\mathbf{u}) + \eta \\
 \tilde{f}_{AM-AM}(z) &= \text{identificationMagnitudeModel}(\|\mathbf{u}\|, \|\mathbf{y}_\eta\|) \\
 W^{\frac{1}{2}} &= \text{diag} \left(\left| \frac{\partial \tilde{f}_{AM-AM}(z)}{\partial z} \right|_{z=|u_1|}, \dots, \left| \frac{\partial \tilde{f}_{AM-AM}(z)}{\partial z} \right|_{z=|u_{N_x}|} \right) \\
 Z &= \left[\Psi_1 \left(\frac{\mathbf{y}_\eta}{K_\gamma} \right), \dots, \Psi_{NDPD} \left(\frac{\mathbf{y}_\eta}{K_\gamma} \right) \right] \\
 Z_w &= W^{\frac{1}{2}} Z \\
 u_w &= W^{\frac{1}{2}} u \\
 \theta &= (Z_w^H Z_w)^{-1} Z_w^H u_w
 \end{aligned} \tag{44}$$

The algorithm to identify the magnitude model is explained in a subsequent paragraph. The identified magnitude model is $\tilde{f}_{AM-AM}(z)$ and $W^{\frac{1}{2}}$ is the $N_x \times N_x$ weighting matrix.

5.2.2 Derivation

To motivate WILA the nonlinear optimization problem is approximated by a least-squares problem for a real function. The basic idea to derive the optimization problem is to calculate the Taylor expansion of the power amplifier model around the ideal predistortion and then neglect higher than second-order error terms.

In order to simplify the derivation the problem is scaled such that the desired gain is one. Therefore

the desired linear gain, the measurement and the power amplifier model have to be scaled:

$$\begin{aligned}\overline{K}_\gamma &= \frac{K_\gamma}{K_\gamma} = 1 \\ \overline{y}_\eta &= \frac{y_\eta}{K_\gamma} \\ \overline{f}_{PA} &= \frac{f_{PA}}{K_\gamma}.\end{aligned}$$

In the following the overline is omitted. If not otherwise mentioned it is referred to the scaled version.

For a desired linear gain of one, the ideal predistortion function f_{DPD}^* is defined indirectly by

$$f_{PA}(f_{DPD}^*(x)) = x. \quad (45)$$

The ideal predistortion can be expressed in terms of the inverse function

$$f_{DPD}^*(x) = f_{PA}^{-1}(x), \quad (46)$$

therefore the ideal predistortion is identical to the inverse power amplifier function.

The linear basis function model subspace for the predistortion is given by H , which is spanned by the basis functions $\Psi_k(x)$. The linear basis function model for the predistortion is expressed by

$$f_{DPD}(x) = \sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(x). \quad (47)$$

A non-ideal predistortion can be decomposed in an ideal predistortion and a predistortion identification error

$$f_{DPD}(x) = f_{DPD}^*(x) + \epsilon_{DPD}(x), \quad (48)$$

therefore the predistortion identification error can be expressed in terms of the linear basis function model and the inverse power amplifier function

$$\begin{aligned}\epsilon_{DPD}(x) &= \sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(x) - f_{DPD}^*(x) \\ &= \sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(x) - f_{PA}^{-1}(x).\end{aligned} \quad (49)$$

The Taylor expansion around the ideal predistortion is

$$\begin{aligned}f_{PA}(f_{DPD}(x)) - x &= f_{PA}(f_{DPD}^*(x) + \epsilon_{DPD}(x)) - x \\ &= f_{PA}(f_{PA}^{-1}(x) + \epsilon_{DPD}(x)) - x \\ &= f_{PA}(f_{PA}^{-1}(x)) + \sum_{n=1}^{\infty} \frac{f_{PA}^{(n)}(f_{PA}^{-1}(x))}{n!} \epsilon_{DPD}(x)^n - x \\ &= \sum_{n=1}^{\infty} \frac{f_{PA}^{(n)}(f_{PA}^{-1}(x))}{n!} \epsilon_{DPD}(x)^n,\end{aligned} \quad (50)$$

Inserting this expression in the objective function (equation 24) yields

$$J = \int_{I_x} p_x(x) (f_{PA}(f_{DPD}(x)) - x)^2 dx = \int_{I_x} p_x(x) \left(\sum_{n=1}^{\infty} \frac{f_{PA}^{(n)}(f_{PA}^{-1}(x))}{n!} \epsilon_{DPD}(x)^n \right)^2 dx. \quad (51)$$

Truncating the series to error terms with an order less or equal to one, the equation simplifies to

$$J = \int_{I_x} p_x(x) \cdot \left(f_{PA}^{(1)}(f_{PA}^{-1}(x)) \right)^2 (\epsilon_{DPD}(x))^2 dx, \quad (52)$$

or in terms of the linear basis function model

$$J = \int_{I_x} p_x(x) \cdot \left(f_{PA}^{(1)}(f_{PA}^{-1}(x)) \right)^2 \left(\sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(x) - f_{PA}^{-1}(x) \right)^2 dx. \quad (53)$$

We obtained a continuous objective function. The goal is to derive a discrete optimization problem. Therefore the integral has to be approximated by a sum over the measurement signal. The input signal is generated by a random process. If the number of samples is large enough an integral can be approximated by a sum over a random variable as described in Appendix B. According to equation 92

$$\int_{I_x} g(x) dx \approx \frac{1}{N_x} \sum_{i=0}^{N_x-1} \frac{g(x[i])}{p_x(x[i])}, \quad (54)$$

where $p_x(x[i])$ is the probability density function of the integration variable. Therefore

$$J = \frac{1}{N_x} \sum_{i=1}^{N_x} \frac{p_x(x[i])}{p_x(x[i])} \cdot \left(f_{PA}^{(1)}(f_{PA}^{-1}(x[i])) \right)^2 \cdot \left(\sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(x[i]) - f_{PA}^{-1}(x[i]) \right)^2. \quad (55)$$

As the desired weight is the input probability function, the optimization problem gets

$$J = \frac{1}{N_x} \sum_{i=1}^{N_x} \left(f_{PA}^{(1)}(f_{PA}^{-1}(x[i])) \right)^2 \cdot \left(\sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(x[i]) - f_{PA}^{-1}(x[i]) \right)^2. \quad (56)$$

Using a final trick, the WILA objective function is derived. Instead of solving the original DPD optimization problem, the circuit is augmented by a copy of the predistortion and the power amplifier, see Figure. Then the post-replica optimization problem is solved. The postreplica optimization problem is

$$J_y = \int_{I_y} p_y(y) \left(f_{PA}(f_{DPD}(y)) - y \right)^2 dy. \quad (57)$$

This means starting from equation (24) x is substituted by y and p_x is substituted by p_y . If the same steps as before are repeated we arrive at the discrete least-square approximation criteria

$$J = \frac{1}{N_x} \sum_{i=1}^{N_x} \left(f_{PA}^{(1)}(f_{PA}^{-1}(y[i])) \right)^2 \cdot \left(\sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(y[i]) - f_{PA}^{-1}(y[i]) \right)^2. \quad (58)$$

or as $f_{PA}^{-1}(y[i])$ is u

$$J = \sum_{i=1}^{N_x} f_{PA}^{(1)}(u[i])^2 \cdot \left(\sum_{k=1}^{N_{DPD}} \theta_k \Psi_k(y[i]) - u[i] \right)^2. \quad (59)$$

Or in matrix notation, omitting the scalar factor $\frac{1}{N_x}$ (a linear transformation doesn't change the optimal solution),

$$\begin{aligned} J &= \mathbf{e}^T \mathbf{W} \mathbf{e} \\ \mathbf{W} &= \text{diag} \left(f_{PA}^{(1)}(u[1])^2, \dots, f_{PA}^{(1)}(u[N_x])^2 \right) \\ \mathbf{e} &= \mathbf{M} \boldsymbol{\theta} - \mathbf{u} \\ \mathbf{M} &= [\Psi_1(\mathbf{y}), \dots, \Psi_{N_{DPD}}(\mathbf{y})] \\ \boldsymbol{\theta} &= [\theta_1, \dots, \theta_{N_{DPD}}]^T \\ \mathbf{u} &= [u[1], \dots, u[N_x]]^T \\ \mathbf{y} &= [y[1], \dots, y[N_x]]^T. \end{aligned} \quad (60)$$

The optimal solution of a weighted least-square problem

$$\min_{\mathbf{x}} \mathbf{e}^T \mathbf{W} \mathbf{e} = \min_{\mathbf{x}} (\mathbf{M} \mathbf{x} - \mathbf{y})^T \mathbf{W} (\mathbf{M} \mathbf{x} - \mathbf{y}) \quad (61)$$

is

$$\mathbf{x}^* = (\mathbf{M}^T \mathbf{W} \mathbf{M})^{-1} \mathbf{M}^T \mathbf{W} \mathbf{y}. \quad (62)$$

The optimal solution can be obtained from the standard least-square solution by substituting M with $M_w = W^{\frac{1}{2}} M$ and y with $y_w = W^{\frac{1}{2}} y$

$$\mathbf{x}^* = (\mathbf{M}_w^T \mathbf{M}_w)^{-1} \mathbf{M}_w^T \mathbf{y}_w. \quad (63)$$

For a complex valued function the complex least-square solution is

$$\mathbf{x}^* = (\mathbf{M}_w^H \mathbf{M}_w)^{-1} \mathbf{M}_w^H \mathbf{y}_w, \quad (64)$$

where $(\cdot)^H$ denotes the Hermitian transpose. The complex equivalent to

$$f_{PA}^{(1)}(u)^2 \quad (65)$$

is

$$\| Df_{PA}|_u \|^2. \quad (66)$$

If the power amplifier function is represented in polar coordinates

$$f_{PA} = f(z) e^{i(g(z)+\varphi)}, \quad (67)$$

the Jacobian is

$$Df_{PA,p} = \begin{bmatrix} \frac{\partial f(z)}{\partial z} & \frac{\partial f(z)}{\partial \varphi} \\ \frac{\partial i(g(z)+\varphi)}{\partial z} & \frac{\partial i(g(z)+\varphi)}{\partial \varphi} \end{bmatrix} = \begin{bmatrix} \frac{\partial f(z)}{\partial z} & 0 \\ i \frac{\partial g(z)}{\partial z} & i \end{bmatrix} \quad (68)$$

and therefore the magnitude gain can be expressed as

$$\| Df_{PA}|_{z=\|u\|} \| = \left\| \left. \frac{\partial f(z)}{\partial z} \right|_{z=\|u\|} \right\|. \quad (69)$$

In the following $f(z)$ will be called the magnitude function or the AM-AM function

$$f(z) = f_{AM-AM}(z). \quad (70)$$

Combining the results above yields the identification algorithm, given in equation 44.

5.2.3 Magnitude model identification

For the magnitude model identification the polynomial basis functions, described in 2.2.3 and the least-squares identification technique described in 2.3 are used. As the magnitude model is used to calculate the derivative, it is not primarily important to get a small error but to obtain a smooth function.

5.3 Model-based indirect learning architecture (MILA)

In [20] it is suggested that forward modeling, generating a noise-free data set and applying ILA mitigates the noise bias problem best. As estimating a model introduces an additional error it is investigated how the MILA error is composed of modeling error and predistortion identification error. On one hand if ILA or WILA are applied to noisy data the solution will be biased, this problem can be avoided by estimating a model from the data first, which is an unbiased estimate, and then generating noise free data from the model. On the other hand estimating a model introduces an additional modeling error in the estimation process. The question is, for which noise level the two errors are balanced.

In the following the MILA algorithm is given, then it is shown how the error can be decomposed in a modeling error and predistortion identification error and finally it is explained how the model can be identified.

5.3.1 Algorithm

Assuming an initial guess for the predistortion is given, the MILA algorithm is:

$$\begin{aligned}\mathbf{u} &= f_{DPD}(\mathbf{x}, \theta_{init}) \\ \mathbf{y}_\eta &= f_{PA}(\mathbf{u}) + \eta \\ f_M &= \text{modelIdentification}(\mathbf{u}, \mathbf{y}_\eta)\end{aligned}\quad (71)$$

$$\begin{aligned}\tilde{\mathbf{y}} &= f_M(\mathbf{u}) \\ \theta &= \text{identILA}(\mathbf{u}, \tilde{\mathbf{y}}) \text{ or } \text{identWILA}(\mathbf{u}, \tilde{\mathbf{y}}).\end{aligned}\quad (72)$$

The predistortion can be estimated either with ILA or WILA, the measurement \mathbf{y}_η is replaced with the model output $\tilde{\mathbf{y}}$.

5.3.2 Derivation, error and model identification

In the following it is investigated how the overall error is composed of the model estimation error $\epsilon_{f_M}(x)$ and the predistortion identification error $\epsilon_{f_{DPD}}(x)$, where,

$$\epsilon_{M(x)} = f_{PA}(x) - f_M(x), \quad (73)$$

and

$$\epsilon_{DPD}(x) = f_M(f_{DPD}(x)) - \gamma(x). \quad (74)$$

Inserting the first term in the objective function yields,

$$\begin{aligned}J &= \int_{I_x} w(x) |f_{PA}(f_{DPD}(x)) - \gamma(x)|^2 dx = \int_{I_x} w(x) |[f_M + \epsilon_M](f_{DPD}(x)) - \gamma(x)|^2 dx \\ &= \int_{I_x} w(x) |f_M((f_{DPD}(x)) - \gamma(x) + \epsilon_M(f_{DPD}(x)))|^2 dx \\ &= \int_{I_x} w(x) |\epsilon_{DPD}(x) + \epsilon_M(f_{DPD}(x))|^2 dx.\end{aligned}\quad (75)$$

An upper bound for the objective value is

$$J \leq 2 \int_{I_x} w(x) |\epsilon_M(f_{DPD}(x))|^2 dx + 2 \int_{I_x} w(x) |\epsilon_{DPD}(x)|^2 dx. \quad (76)$$

To minimize the model error contribution to the upper bound, the following objective function has to be minimized

$$J_M = \int_{I_x} w(x) |\epsilon_M(f_{DPD}(x))|^2 dx. \quad (77)$$

Therefore to identify a model for the predistortion identification an initial guess of the predistortion is required. Rewriting the objective function in terms of a sum instead of an integral (see Appendix B) and substituting $f_{DPD}(x)$ with u yields the least-square objective function for the model identification

$$J_M = \frac{w(x)}{p_x(x)} \sum_{i=1}^{N_x} |\epsilon_{PA}(f_{DPD}(x[i]))|^2 = \sum_{i=1}^{N_x} |\epsilon_{PA}(u[i])|^2 = \sum_{i=1}^{N_x} \left| \sum_{k=1}^{N_{PA}} a_k \Phi_k(u[i]) - y[i] \right|^2. \quad (78)$$

In this thesis the basis functions and the least-square algorithm described in the subsection 2.2.3 and 2.3 to identify the model.

Later on the modeling and predistortion identification error contributions will be discussed. In terms of NMSE the modeling error $NMSE_M$ is given by

$$\begin{aligned}NMSE_M &= 10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} |f_{PA}(u[n]) - f_M(u[n])|^2}{\sum_{n=0}^{N-1} |f_{PA}(u[n])|^2} \right) \\ &= 10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} |y[n] - f_M(u[n])|^2}{\sum_{n=0}^{N-1} |y[n]|^2} \right),\end{aligned}\quad (79)$$

and the predistortion identification error is given by

$$NMSE_{DPD} = 10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} |f_M(u[n]) - \gamma(x[n])|^2}{\sum_{n=0}^{N-1} |f_M(u[n])|^2} \right). \quad (80)$$

The composed error is denoted by $NMSE_{MILA}$.

6 Results and Discussion

The result section is structured in two parts. In the first part a single example is selected to show vividly the effect of the suggested improvements. In the second part the experiments are repeated for a larger testbench and the resulting NMSE and ACPR is presented in order to show that the results are valid not only for a good testcase choice.

6.1 Illustrating example

6.1.1 Suggestion 1: Reinforcement data grid for the initial guess

Figure 31 and 32 show the AM/AM plot of the identified predistortion after the initial guess the first and the second iteration of ILA with and without reinforcement grid (SNR ≈ 40 dB). For a large backoff (see Figure 31) the extrapolation problem is not severe because the ring-shaped area without data points is small (take a look at the extrapolation point). But for a decreased backoff (see Figure 32), the predistortion is extrapolated over a larger region. If the predistortion is overestimated, the extrapolation doesn't make the results worse. Because the too large signal is clipped and therefore for the first iteration the whole area is covered with data points. If the predistortion is underestimated iterative ILA converges anyway but instead of one iteration, at least three iterations are necessary until the predistortion doesn't change visible anymore.

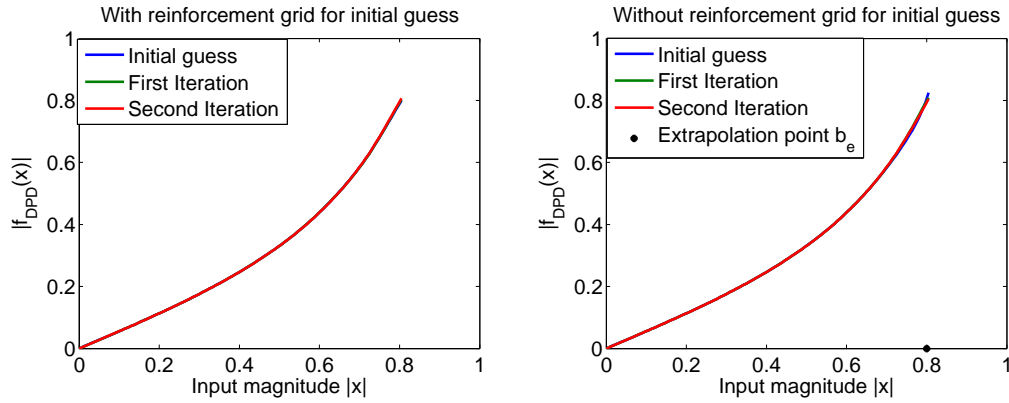


Figure 31: Comparison convergence of the predistortion function with and without reinforcement data grid for a backoff of 14.37 dB

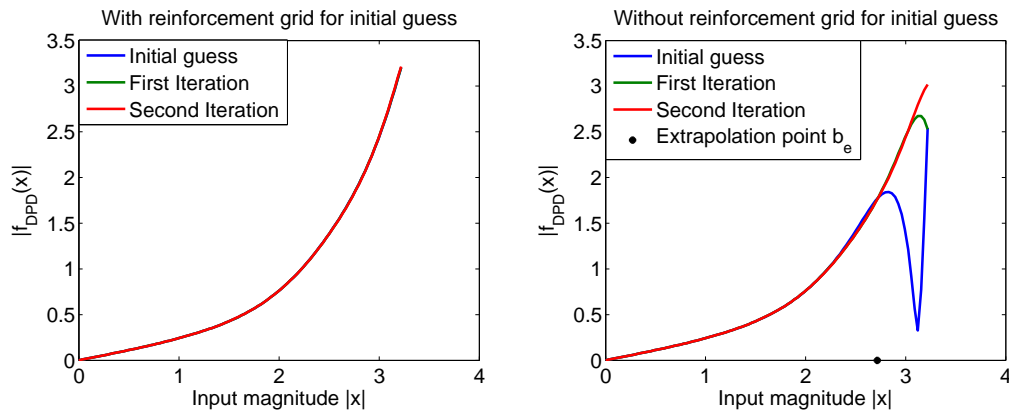


Figure 32: Comparison convergence of the predistortion function with and without reinforcement data grid for a backoff of 10.42 dB. The first estimate was underestimated.

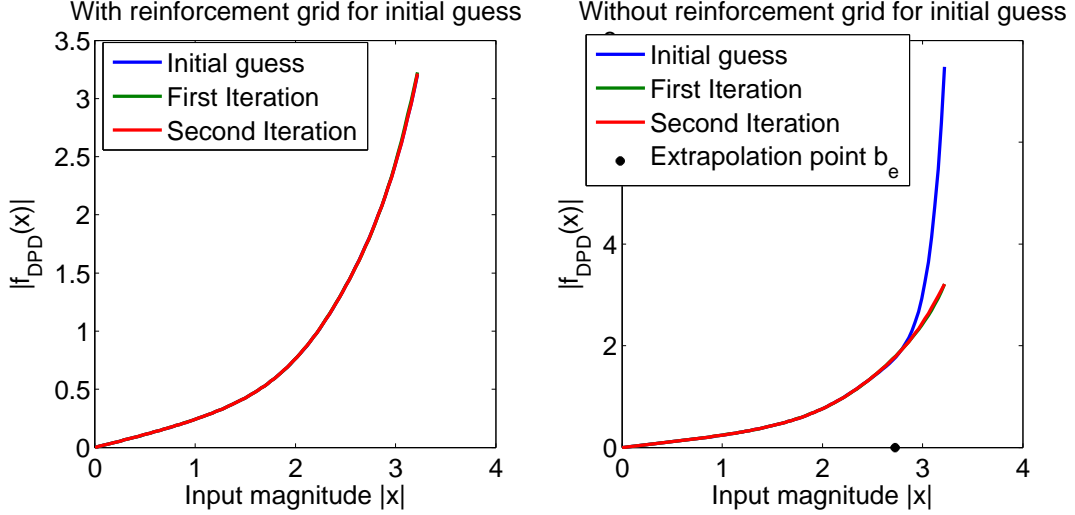


Figure 33: Comparison convergence of the predistortion function with and without reinforcement data grid for a backoff of 10.42 dB. The first estimate was overestimated.

6.1.2 Suggestion 2: Weighted indirect learning architecture

In the cascade order problem description it was illustrated that for ILA the envelope of we^2 doesn't look as expected, see Figure 26. In comparison for WILA the envelope of we^2 looks more as expected: flat in the middle and slightly higher at the borders, see Figure 34.

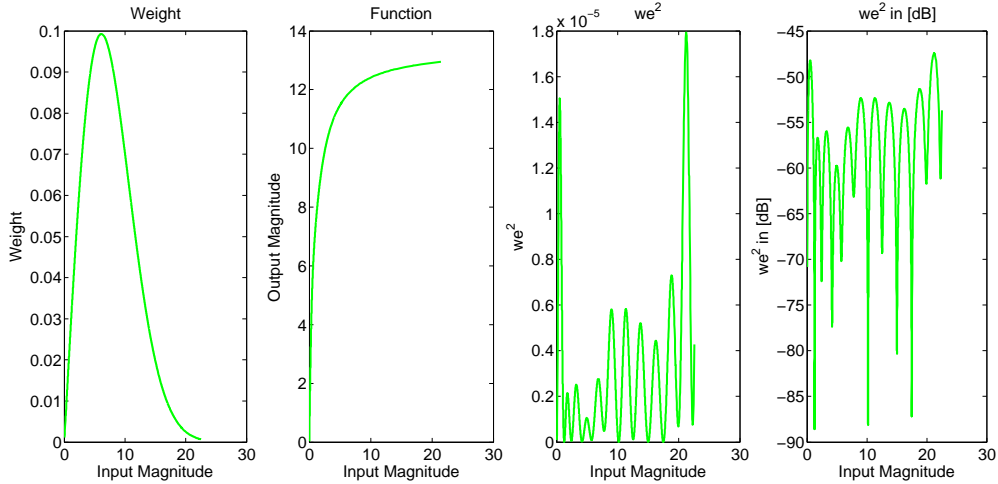


Figure 34: Weight, Function and we^2 for an WILA solution

Figure 35, 36 and 37 show the improvement of WILA compared to ILA for three different backoffs. The first column shows the power amplifier AM/AM plot, the linearization results for ILA and WILA and the scaled input data distribution. The second column shows the magnitude error

$$|e| = |f_{PA}(f_{DPD}(x)) - K_{\gamma}x| \quad (81)$$

for ILA and WILA and the expected magnitude error envelope shape for the optimal solution. The third column shows the squared error $|e|^2$ in dB ($10 \log_{10}$) and the expected squared error envelope shape for the optimal solution.

The first observation is that WILA improves the results. The error shape matches better with the expected error envelope shape of the optimal solution and the mean-squared error is decreased. For a very small backoff the effect is even visible in the linearization plot. While the ILA linearization has a strange hill for small input magnitude values the WILA linearization looks linear. In Figure 37 for

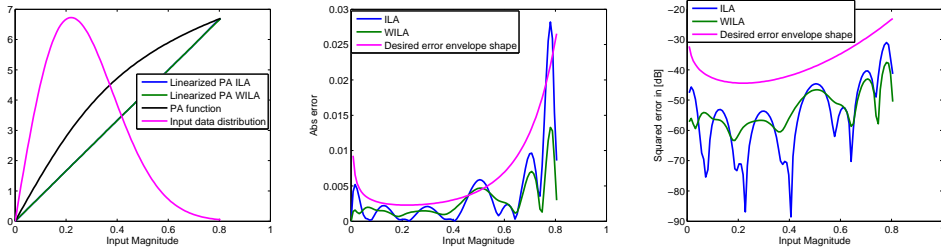


Figure 35: Comparison ideal predistortion and estimated predistortion squared-error for ILA and WILA, backoff 14.37 dB

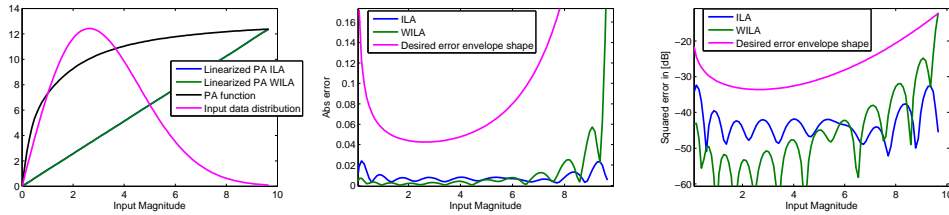


Figure 36: Comparison ideal predistortion and estimated predistortion squared-error for ILA and WILA, backoff 9.03 dB

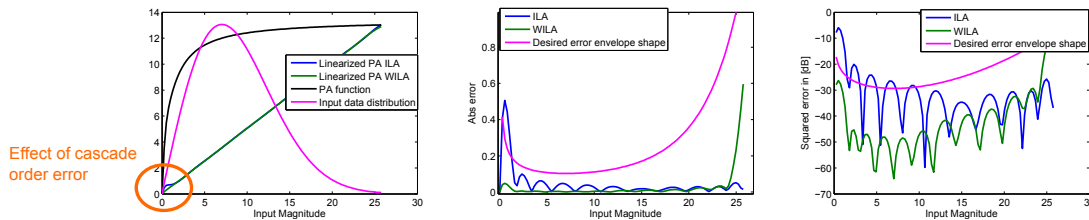


Figure 37: Comparison ideal predistortion and estimated predistortion squared-error for ILA and WILA,, backoff 8.59 dB

the most probable input magnitude values (≈ 7) the WILA linearization is 20 dB better than the ILA linearization.

The second observation is, that the difference between ILA and WILA increases with decreased backoff. If the power amplifier function would be linear the weighting function (the derivative of the AM model) would be a constant and wouldn't influence the result. The smaller the backoff the more varies the derivative of the magnitude model and the larger is the effect of the weighting function.

6.2 Testbench

For the testbench, the Arctan model, see equation 2, was selected as power amplifier model. The different testcases are created by varying the backoff and the noise level. In each plot the NMSE and the difference between ACPR of the output and input signal (see subsection 2.4) are plotted versus the maximum input magnitude for a constant input noise power level. The ACPR of the input signal is not $-\infty$ [dB], because the pulse-shaping filter is a non-ideal bandpassfilter. Figure 38 a) shows the connection between maximum input power level and backoff for the ideal predistortion. The advantage of plotting the NMSE and ACPR against the maximum input magnitude and not against the backoff is, that it is not changed by the predistortion estimate. The backoff is changed by the predistortion estimate, for example, if a wrong estimate would yield a predistortion whose output is always the maximum power amplifier magnitude the backoff would be zero.

As different backoffs correspond to different output noise levels the SNR varies within each plot, even so the input noise power is constant. Therefore in addition to the SNR the peak-signal-to-noise power (PSNR) as defined in equation 94 is given. The PSNR is defined as the ratio between maximum

output power and noise power.

In order to give a more intuitive impression of the selected backoffs, maximum input noise level and noise levels, in Figure 38 a) the backoff is plotted versus the maximum input magnitude and in Figure 38 b) different noise samples are shown.

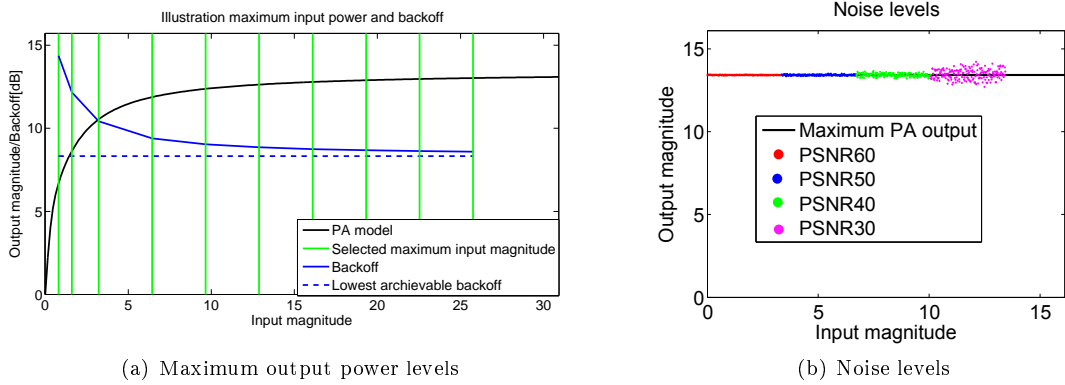


Figure 38: Illustration noise levels and maximum output power levels

With lowest achievable backoff, it is referred to the backoff for which the maximum input magnitude is mapped to the maximum PA output magnitude. The backoff can be decreased below that level only by compressing the input values and therefore by not linearizing the PA. The lowest achievable backoff can be decreased by using crest-factor reduction schemes on the input data.

6.2.1 Suggestion 1: Reinforcement data grid for the initial guess

Figure 39 and 40 show the NMSE for the initial guess, after the first and second iteration plotted versus the maximum input magnitude. The experiment was repeated ten times with different input and noise realizations. In Figure 39 the SNR is 50 ± 2 dB and in Figure 40 the SNR is 30 ± 2 dB.

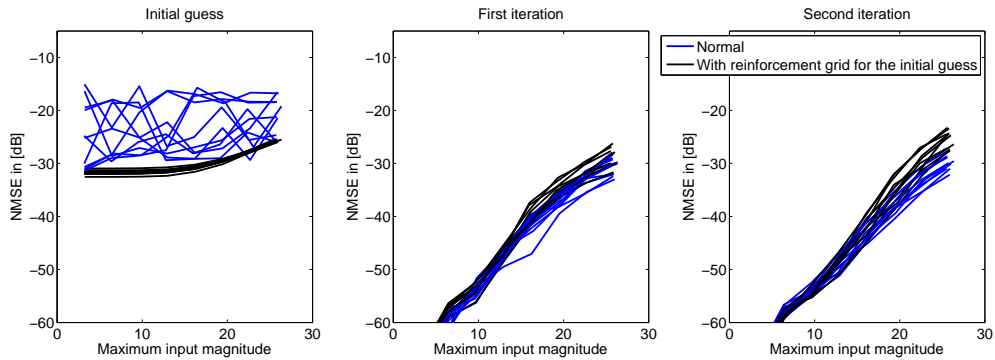


Figure 39: ILA: Initial guess with and without reinforcement grid, SNR between 49.34 and 51.45 dB, PSNR 60 dB

Discussion For the initial guess the error fluctuates strongly without a reinforcement grid. For an SNR of 50 dB the NMSE with and without reinforcement grid is indistinguishable after the first iteration and for an SNR of 30 dB the NMSE is indistinguishable after the second iteration.

In most cases the predistortion function is overestimated for large input magnitude values during the initial guess, and therefore in the first iteration the complete linearization area is covered with data points.

It is also visible that for an increased noise level, the NMSE varies more between the different experiments.

Further the plots show that after the first iteration the results don't improve significantly, and one

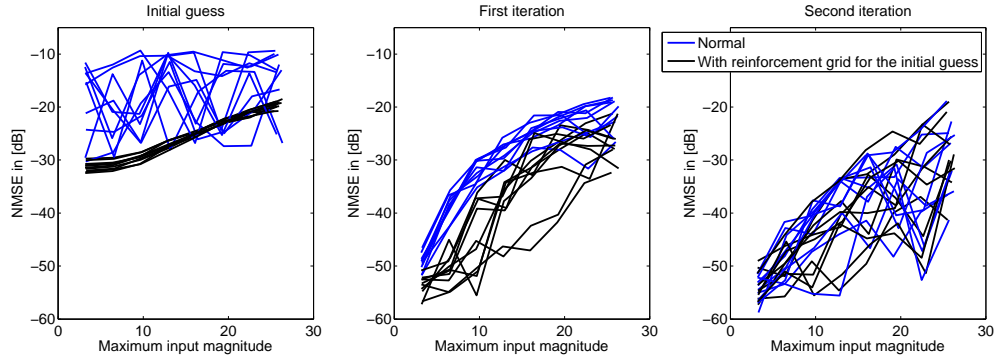


Figure 40: ILA: Initial guess with and without reinforcement grid, SNR between 29.30 and 31.48 dB, PSNR 40 dB

iteration of ILA seems to be enough to obtain the predistortion estimate.

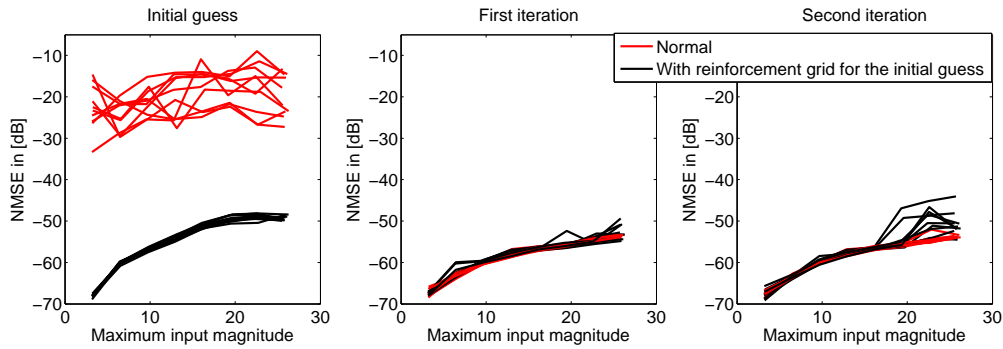


Figure 41: WILA: Initial guess with and without reinforcement grid, SNR between 49.34 and 51.45 dB, PSNR 60 dB

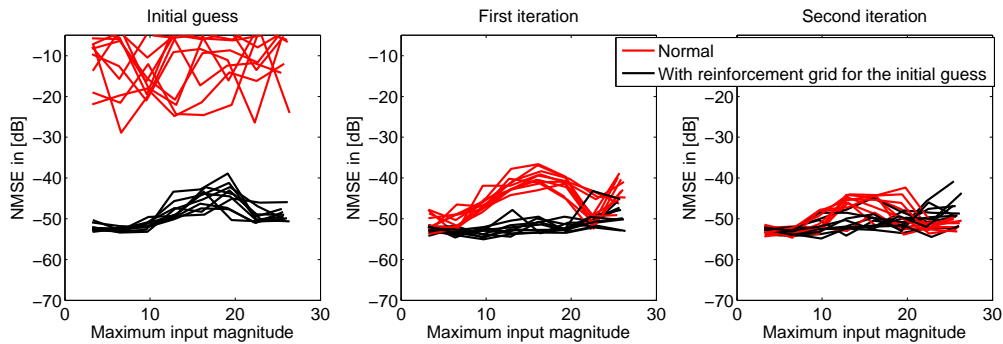


Figure 42: WILA: Initial guess with and without reinforcement grid, SNR between 29.30 and 31.48 dB, PSNR 40 dB

The same results were obtained, when the initial guess and the iterations were calculated with WILA, see Figure 41 and 42.

Conclusion The results have shown, that the predistortion is extrapolated for the initial guess. The effect is larger for a lower backoff. For a low noise level the predistortion quality was similar with and without reinforcement data grid after the first iteration. For a higher noise level this was the case after two iterations.

As it seems that the effect of the extrapolation error is mitigated through multiple iterations, we would

say that it is not necessary to add an extra reinforcement data grid for the initial guess. We think an exception should be made when the predistortion identification algorithm is used without the surveillance of a human being. Further as the extrapolation can yield very high power amplifier input values, the signal has to be clipped to the maximum allowable input power before the power amplifier.

6.2.2 Suggestion 2: Weighted indirect learning architecture

As the results vary especially for higher noise levels with different input data and noise realizations, the results in this section are obtained by repeating each experiment ten times and taking the mean value m . Further the standard deviation $m + \sigma$ and $m - \sigma$ is plotted. The mean and the standard deviation are calculated for the logarithmic values. For the ACPR the mean and standard deviation is calculated after the ACPR of the input signal x was subtracted

$$ACPR_{mean} = \frac{1}{N_i} \sum_{i=1}^{N_i} (ACPR_y(i) - ACPR_x(i)). \quad (82)$$

The initial guess is calculated with ILA. Within one experiment i the same initial guess, the same noise variance and the same input data realization is used.

Figure 43 shows the mean NMSE for different noise levels and linearization areas. Figure 44 shows the mean ACPR for different noise levels and linearization areas.

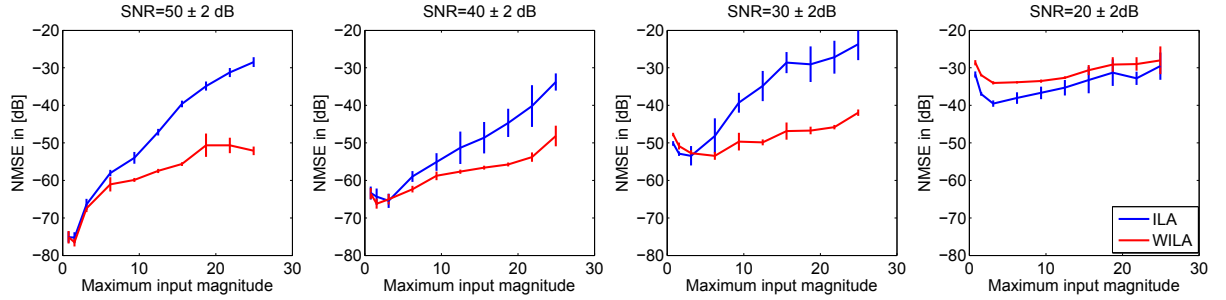


Figure 43: Comparison ILA and WILA, NMSE for different noise levels and linearization areas

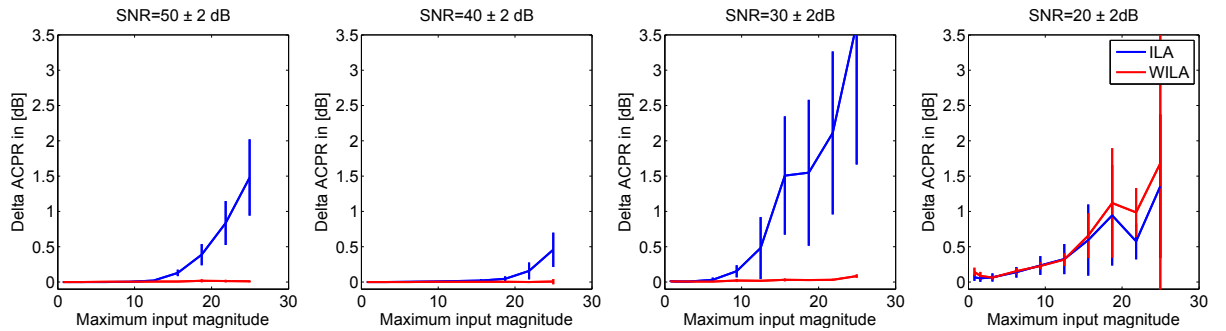


Figure 44: Comparison ILA and WILA, ACPR for different noise levels and linearization areas

Discussion Figure 43 and figure 44 show that WILA has a smaller NMSE and ACPR as well as a smaller standard deviation up to an SNR of 30 dB. For an SNR of 20 dB ILA has a smaller SNR and both techniques have approximately the same ACPR. That WILA has a clearly better NMSE for small noise levels shows that the derivation of WILA was correct and WILA is the better least-square approximation of the nonlinear predistortion identification problem. The larger error for an SNR of 20 dB can be explained by a bad magnitude model derivative identification. We took the first idea to identify the derivative from noisy data and didn't dig into finding the best technique. So there is a good chance to improve WILA further for noisy data by using a better derivative identification technique.

Conclusion The results show that we found a technique to avoid the cascade order problem, which increases the computational complexity only by a factor of two. The technique could be further improved by investigating if there is a better technique to identify the derivative of the magnitude model from noisy data. The technique could be simplified by estimating the weight in advance, for example for a batch of power amplifiers which have only small fabrication differences.

6.2.3 Suggestion 3: Model-based indirect learning architecture

As the results in the last section show that for the noise-free case WILA yields the better identification results in this section WILA is taken as basis for the model-based indirect learning architecture. This means that after a noise free data set was generated from the model, WILA is used for the predistortion identification. The results are either compared to ILA or WILA depending on which technique did yield the better results in the last section. The plots were obtained in a similar fashion as in the last section by averaging over ten experiments. Figure 45 shows the NMSE and Figure 46 shows the ACPR.

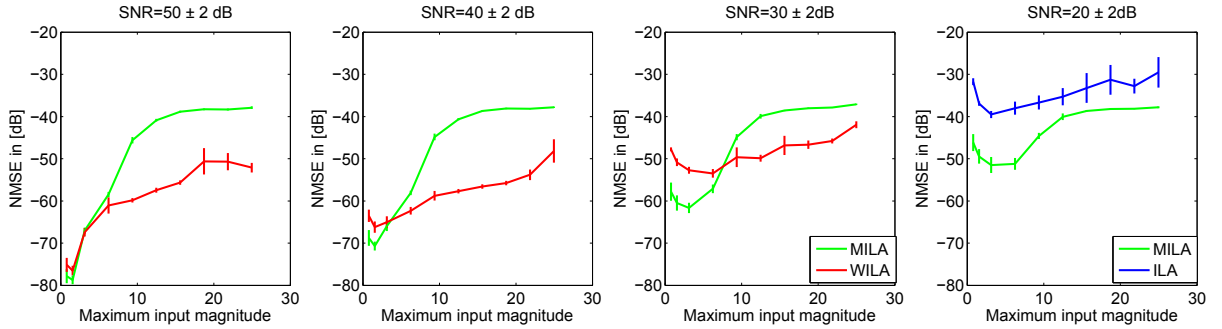


Figure 45: Comparison MILA and ILA/WILA, NMSE for different noise levels and linearization areas

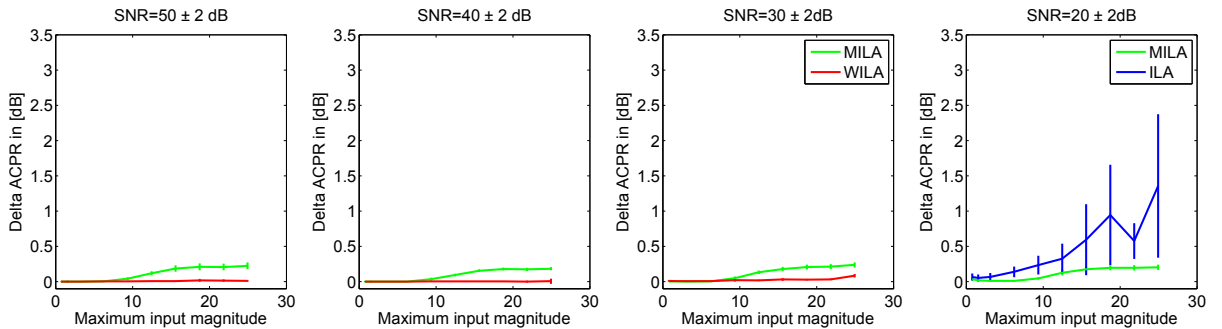


Figure 46: Comparison MILA and ILA/WILA, ACPR for different noise levels and linearization areas

In section 5.3.2 we showed that the MILA error can be decomposed in a modeling and a predistortion identification contribution. Figure 47 shows how the MILA error is composed of modeling error and predistortion identification error.

Discussion As expected there is a noise level for which MILA performs better than the other techniques, in our experiments this was the case for an SNR of 20 dB. Figure 45 and Figure 46 show that MILA improves the predistortion identification for a large backoff and an SNR of 30 dB and independent of the backoff for an SNR of 20 dB. While the NMSE of WILA increases with the increased noise level, the NMSE of MILA seems constant at a high level. Figure 47 shows that the composed MILA error is dominated by the model identification error. The model can be improved by using more data points for the model identification. The question is if a comparison with different amounts of data is fair. An upper bound for the predistortion error contribution is twice the predistortion identification error (see equation 76), therefore the model identification error should be smaller than the expected noise-free predistortion identification error plus 3 dB. In Figure 47 the NMSE is higher than in the noise-free case in Figure 43. The explanation is that the estimated model doesn't need to be monotone. The best

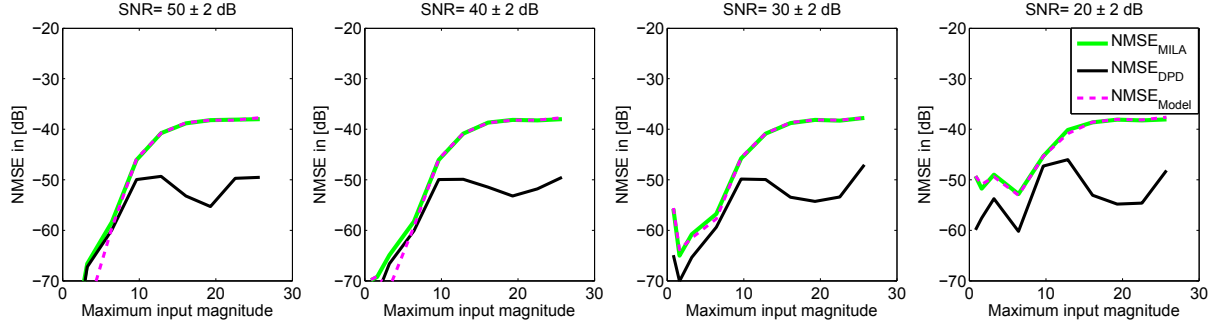


Figure 47: Error composition MILA, squared error, model identification error and predistortion identification error

possible predistortion for a not monotone model does linearize the power amplifier less good than the best possible predistortion for a monotone model.

Conclusion For noise levels smaller than an SNR of 30 dB, it is possible to estimate a sufficiently good predistortion $\text{NMSE} < -45$ dB without a model-based technique. Many papers dealing with model-based predistortion identification techniques only consider the predistortion identification part. In this section it was shown that the modeling part is equally important. A good rule of thumb for the modeling error is, that it should be smaller than the predistortion identification error of the considered technique plus 3 dB. Further it is recommended to add reinforcement data to the model identification data set if the data density is scarce at the border of the linearization area, as can for example be seen in Figure 24. Finally the model-based technique can only be used after an initial guess of the predistortion was already calculated with either ILA or WILA, because the model should minimize

$$10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} |y[n] - f_M(f_{DPD}(x[n]))|^2}{\sum_{n=0}^{N-1} |y[n]|^2} \right) \quad (83)$$

and not

$$10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} |y[n] - f_M(x[n])|^2}{\sum_{n=0}^{N-1} |y[n]|^2} \right). \quad (84)$$

6.3 Summary conclusion

For all testcases it was possible to improve the ILA identification results in terms of NMSE and ACPR. For testcases with a low backoff and low noise level the results were most improved by WILA. For a high backoff and high noise level the results were most improved by MILA. For a low backoff the cascade order error prevents good identification results with ILA and it is recommended to use WILA instead.

In [29] two drawbacks of ILA the bias due to noise and the problem that originates from changing the cascade order between pre- and postdistortion are discussed. They mention that so far for the second problem only solutions which increase the computational complexity by a very significant factor were proposed. In this thesis a new approach of addressing the cascade order problem that increases the computational complexity only by a factor of two is proposed. The simulations show that significant improvements in terms of ACPR and NMSE can be archived with the new technique.

In [20] Morgan et al. propose a technique to address the noise bias problem. Their technique, to first identify a model and then create a noise-free predistortion identification data set, was further investigated in this thesis. It was shown that the overall error can be decomposed in a model identification and a predistortion identification contribution. A good rule of thumb for the model identification quality is, that the model identification error, as defined in equation 79, should have the same order of magnitude as the noise-free predistortion identification error. Therefore for our experiments the model identification error should be (depending on the backoff) somewhere between -70 and -50 dB. In our experiments the modeling error dominated the overall MILA error. One way to obtain a better model

is to increase the number of identification samples. For example in [20] 200,000 samples are used for forward modeling while only 40,000 samples are used for the inverse modeling. That more samples were used for the forward modeling than for the predistortion identification could also explain, why in their experiments MILA improves the result for an SNR of 35 dB, while in our experiments MILA improves the results only for an SNR of 20 dB.

Many direct learning architectures (DLA) [29] require a model. Either because they are gradient-based or because they require many function evaluations. It was shown that the modeling error can dominate the overall error. For a model-based technique a more fancy predistortion identification technique can only improve the results if the model is good enough such that the predistortion error dominates the composed error.

Finally it was shown that the predistortion is extrapolated after the initial guess. The extrapolation error seems to be mitigated quite well through further iterations. As the extrapolation error can yield very high input values, the input magnitude should be clipped before the power amplifier. For unsurveilled automatic identification we would recommend to use an reinforcement grid and to avoid the extrapolation for the initial guess.

7 Future work

For complex power amplifier models WILA showed promising results. But as mentioned in the power amplifier modeling part, many power amplifiers exhibit memory effects. Therefore the method has to be extended to a memory-based case. A simple way to extend the method to a memory-based case is to neglect the memory effects for the calculation of the weight. It has to be verified that the same results are obtained for a memory-based case and that the same conclusions can be drawn for a comparison of ILA/WILA and WILA/MILA. After that the results have to be verified with measurement data.

If this investigation shows positive results it could be investigated how WILA can be incorporated in an adaptive algorithm in order to track changing operating conditions. A very simple fashion to adapt the predistortion is to regularly save a new data block and to identify and update the predistortion parameters accordingly. But as it is not clear how much the power amplifier is influenced by ambient conditions a prestudy might be interesting. In the prestudy it could be evaluated how much the power amplifier is influenced by operating conditions like the mean output power or the ambient temperature.

Another important and missing investigation is to compare memory polynomial WILA to a look-up table based identification technique. LUTs are the dominating predistortions in an industrial environment, as mentioned in the presentations [24] and [27].

A related, relevant, but far more complicated problem is the predistortion identification for MIMO systems. To steer the electromagnetic beam more accurately or in order to use multiple cheaper transmitter architectures instead of a single expansive and high-power one a research trend is to investigate transmitter arrays. The identification problem gets more difficult because there is cross-talk between the different antennas. An antenna acts always as both as a transmitter and as a receiver.

Appendices

A Probability density distributions

If the real as well as the imaginary part are Gaussian distributed

$$g(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \quad (85)$$

with zero mean $\mu = 0$ the corresponding absolute value is Rayleigh distributed

$$r(x; \sigma^2) = \frac{x}{\sigma^2} e^{-x^2/2\sigma^2} \quad (86)$$

and the corresponding angle is uniformly distributed

$$u(x; a, b) = \begin{cases} \frac{1}{b-a} & \text{for } x \in [a, b] \\ 0 & \text{otherwise} \end{cases}, \quad (87)$$

with $[a, b] = [-\pi, \pi]$, as

$$p(z, \varphi) = \det \frac{\partial(x, y)}{\partial(z, \varphi)} p(x, y) = z g(x; 0, \sigma^2) \cdot g(y; 0, \sigma^2) = z \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\left(\frac{x^2+y^2}{\sigma^2}\right)} = r(z; \sigma^2) \cdot u(\varphi; -\pi, \pi). \quad (88)$$

B Approximating an integral by a sum of a statistical variable

An integral can be approximated by a finite sum

$$\int_a^b w(x) f(x) dx = \sum_{i=1}^N \Delta\delta_i w(x_i) f(x_i). \quad (89)$$

If \mathbf{x} is a compound event, a set of random input values, each with a probability density function p_x and a cumulative probability function c_x , either for each event the $\Delta\delta_i$ s are calculated by sorting x_i and taking the mean difference to the adjacent points or assuming a large number of input samples, instead of the $\Delta\delta_i$ s the expected value can be used

$$\Delta\delta_E(x) = \frac{1}{N p_x(x)}. \quad (90)$$

The expected interval bandwidth $\Delta\delta_E(x)$ is derived, by dividing the interval length by the expected number of input samples in this interval and taking the limiting value

$$\lim_{|b-a| \rightarrow 0} \frac{b-a}{N(c_x(b) - c_x(a))} = \frac{1}{N c_x^{(1)}(x)} = \frac{1}{N p_x(x)}. \quad (91)$$

Therefore the integral can be approximated by

$$\int_a^b w(x) f(x) dx \approx \frac{1}{N} \sum_{i=1}^N \frac{w(x_i)}{p_x(x_i)} f(x_i) \Big|_{N \gg 1}. \quad (92)$$

C Transformation properties of probability density distributions

If the probability density function of x is p_x and $y = f(x)$, the probability density function of y is

$$p_y(y) = \frac{p_x(f^{-1}(y))}{f^{(1)}(f^{-1}(y))} \quad (93)$$

D Simulation setup and simulation parameters

Figure 48 shows the simulation setup. The input data is generated as described in the subsection *Input data*, then the signal is clipped to the linearization area and predistorted. If the data set is used for the model identification or for the initial guess an extra set of measurement points is added to the identification data set, as explained in more detail in the subsection *additional identification data set*. Then the signal is amplified. Then the output signal is measured, which means that measurement noise is added to the signal. The PA input u and the measured output y_η are used to identify the predistortion parameters θ . Compared to Figure 13, the DAC/ADC blocks and the up-and downsampling are missing. This components are assumed to be ideal.

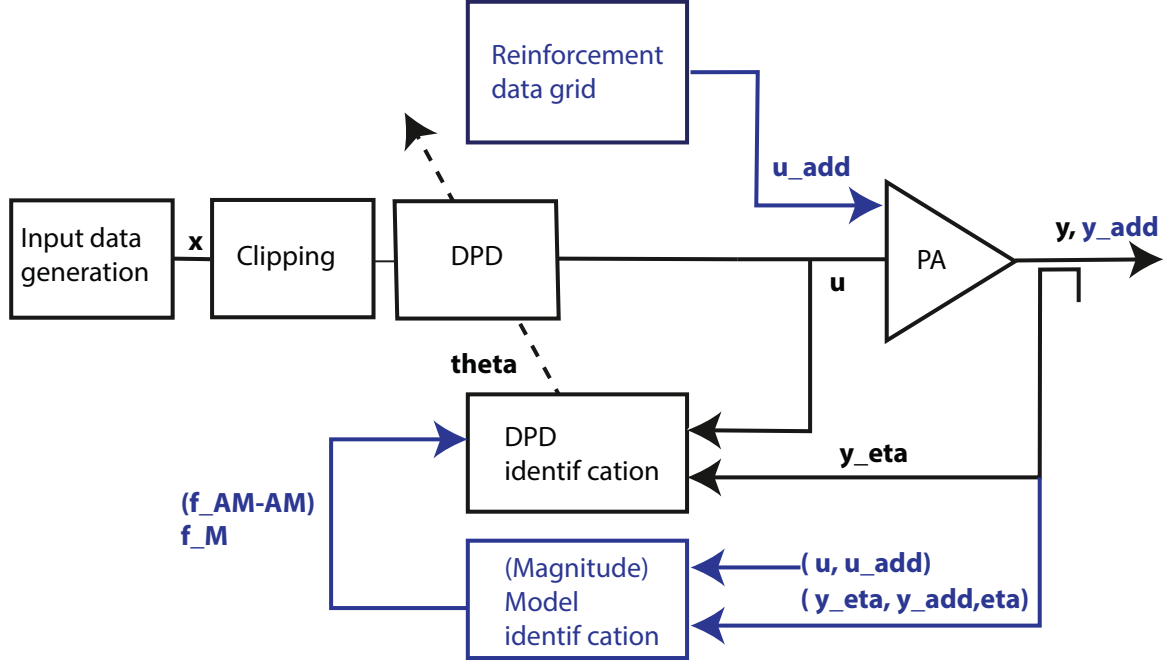


Figure 48: Memoryless simulation setup

Simulation parameters Table 6 summarizes the simulation parameters. Several simulation parameters are substituted by other expressions in order to make them comparable for different PA functions, see Table 7. The noise variance is substituted with the peak-signal-to-noise ratio (PSNR) and the linearization area is substituted by the percentage of data contained in the linearization area. In the microwave engineering community it is common to speak of noise levels in terms of signal-to-noise ratios (SNR). As the output signal power changes when the signal is predistorted for different linearization areas, instead of the SNR the PSNR is used. The PSNR is defined as the maximum signal power divided by the noise power

$$PSNR_{dB} = 10 \log_{10} \left(\frac{P_{max}}{P_\eta} \right). \quad (94)$$

The noise power is calculated from the mean and variance of the corresponding Rayleigh distribution

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^{N-1} |\eta|^2 = var(|\eta|) + \mu(|\eta|)^2 = \frac{4 - \pi}{2} \sigma^2 + \left(\sigma \sqrt{\frac{\pi}{2}} \right)^2 = 2\sigma^2, \quad (95)$$

for complex white Gaussian noise the PSNR is given by

$$PSNR_{dB} = 10 \log_{10} \left(\frac{y_{max}^2}{2\sigma_\eta^2} \right), \quad (96)$$

where

$$y_{max} = \max_{u \in \mathbb{R}^M} |f_{PA}(u)|, \quad (97)$$

or respectively the noise variance is given by

$$\sigma_\eta^2 = \frac{y_{max}^2}{2 \cdot 10^{\frac{PSNR_{dB}}{10}}}. \quad (98)$$

The linearization area is calculated numerically, by generating an input data realization, sorting the data and taking the magnitude of the last sample which lies within the first p-percent of the total number of samples, see `getLinearizationIntervalFromPercentage()`.

Table 6: Simulation parameters

Name	Function	Subsection
N_x	Number of data samples	Input data generation
N_v	Number of evaluation data samples	Performance metrics
σ_x	Standard deviation of complex input data	Input data generation
f_c	Normalized bandpassfilter cutoff frequency $\in (0, 1)$	Input data generation
b_u	Radius linearization area	Linearization area
N_{DPD}	DPD subspace order	DPD subspace
N_{PA}	PA model subspace order	Linear basis function models
σ_η^2	Noise variance	Noise
f_{PA}	PA model	Memoryless power amplifier models
p_L	Specifies area for reinforcement grid	Additional identification data set
N_{extra}	Maximum number of extra data points	Additional identification data set

Table 7: Substituted simulation parameters

Substitute	Original	Function
p_c	I_u	Probability that input sample is contained in linearization area
$PSNR$	σ_η^2	Peak-signal-to-noise ratio

The probability that the data stays unclipped can be specified in the simulation parameter script by the parameter p_c . For the simulation the linearization area is calculated by creating a realization of the input data signal and determining the area that contains p_c -percent of the data.

Comparability of different testcases To ensure comparability between the different testcases, the identification and evaluation data sets and the linearization areas are calculated in advance for a standard deviation of $\sigma_x = 1$ and then scaled by the standard deviation of the input data of the testcase.

References

- [1] *Circularly polarized light*. Wikipedia.
<http://en.wikipedia.org/wiki/File:Circular.Polarization.Circularly.Polarized.Light.With.Components.Right.Handed.svg>.
- [2] *Constellation Diagram PSK*. Wikipedia.
http://en.wikipedia.org/wiki/File:8PSK_Gray_Coded.svg.
- [3] *Constellation Diagram QAM*. Wikipedia.
http://commons.wikimedia.org/wiki/File:16QAM_Gray_Coded.svg.
- [4] *Elementary electric wave*. IBS Japan .
http://www.ibsjapan.co.jp/html/elementary_electric_wave/04-15.html.
- [5] *Frequency Allocation Plan US*. Wikipedia.
http://en.wikipedia.org/wiki/Frequency_allocation.
- [6] *Linearly polarized light*. Wikipedia.
http://en.wikipedia.org/wiki/File:Onde_electromagnetique.svg.
- [7] M. ABI HUSSEIN, V. BOHARA, AND O. VENARD, *On the system level convergence of ILA and DLA for digital predistortion*, in Wireless Communication Systems (ISWCS), 2012 International Symposium on, aug. 2012, pp. 870–874.
- [8] E. ASCHBACHER AND M. RUPP, *Modelling and identification of a nonlinear power-amplifier with memory for nonlinear digital adaptive pre-distortion*, in Signal Processing Advances in Wireless Communications, 2003. SPAWC 2003. 4th IEEE Workshop on, june 2003, pp. 658–662.
- [9] R. N. BRAITHWAITE, *Digital Front-End in Wireless Communications and Broadcasting*, Cambridge University Press, 2011, ch. General principles and design overview of digital predistortion, pp. 144–189.
- [10] H. CAO, H. NEMATI, A. TEHRANI, T. ERIKSSON, J. GRAHN, AND C. FAGER, *Linearization of Efficiency-Optimized Dynamic Load Modulation Transmitter Architectures*, Microwave Theory and Techniques, IEEE Transactions on, 58 (2010), pp. 873–881.
- [11] J. CAVERS, *Optimum table spacing in predistorting amplifier linearizers*, Vehicular Technology, IEEE Transactions on, 48 (1999), pp. 1699–1705.
- [12] L. DING AND Z. ET AL., *A robust digital baseband predistorter constructed using memory polynomials*, Communications, IEEE Transactions on, 52 (2004), pp. 159–165.
- [13] L. DING, Z. MA, D. MORGAN, M. ZIERDT, AND J. PASTALAN, *A least-squares/Newton method for digital predistortion of wideband signals*, Communications, IEEE Transactions on, 54 (2006), pp. 833–840.
- [14] L. DING, G. ZHOU, D. MORGAN, Z. MA, J. KENNEY, J. KIM, AND C. GIARDINA, *Memory polynomial predistorter based on the indirect learning architecture*, in Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE, vol. 1, nov. 2002, pp. 967–971 vol.1.
- [15] C. EUN AND P. ET AL., *A new Volterra predistorter based on the indirect learning architecture*, Signal Processing, IEEE Transactions on, 45 (1997), pp. 223–227.
- [16] C. EUN AND E. POWERS, *A predistorter design for a memory-less nonlinearity preceded by a dynamic linear system*, in Global Telecommunications Conference, 1995. GLOBECOM '95., IEEE, vol. 1, nov 1995, pp. 152–156 vol.1.
- [17] S. HONG, Y. Y. WOO, J. KIM, J. CHA, I. KIM, J. MOON, J. YI, AND B. KIM, *Weighted Polynomial Digital Predistortion for Low Memory Effect Doherty Power Amplifier*, Microwave Theory and Techniques, IEEE Transactions on, 55 (2007), pp. 925–931.

- [18] H. W. KANG, Y. S. CHO, AND D. H. YOUN, *On compensating nonlinear distortions of an OFDM system using an efficient adaptive predistorter*, Communications, IEEE Transactions on, 47 (1999), pp. 522 –526.
- [19] J. LIEBTREU, *Proposed System Impairment Models*. Presentation to IEEE 802.16 Broadband Wireless Access Working Group.
- [20] D. MORGAN, Z. MA, AND L. DING, *Reducing measurement noise effects in digital predistortion of RF power amplifiers*, in Communications, 2003. ICC '03. IEEE International Conference on, vol. 4, may 2003, pp. 2436 – 2439 vol.4.
- [21] D. MORGAN, Z. MA, J. KIM, M. ZIERDT, AND J. PASTALAN, *A Generalized Memory Polynomial Model for Digital Predistortion of RF Power Amplifiers*, Signal Processing, IEEE Transactions on, 54 (2006), pp. 3852 –3860.
- [22] H. QIAN AND G. T. ZHOU, *Digital Front-End in Wireless Communications and Broadcasting*, Cambridge University Press, 2011, ch. Adaptive digital baseband predistortion: design and implementation, pp. 280–308.
- [23] R. RAICH, H. QIAN, AND G. ZHOU, *Orthogonal polynomials for power amplifier modeling and predistorter design*, Vehicular Technology, IEEE Transactions on, 53 (2004), pp. 1468 – 1479.
- [24] D. RÖNNOW, *Comparison of DPD in tele-and satellitecommunication*, in GHz Centre Workshop: Signal Processing and Amplifiers, 2012.
- [25] A. SALEH, *Frequency-Independent and Frequency-Dependent Nonlinear Models of TWT Amplifiers*, Communications, IEEE Transactions on, 29 (1981), pp. 1715 –1720.
- [26] M. SAUTER, *Base Station Numbers*.
http://mobilesociety.typepad.com/mobile_life/2009/06/base-station-numbers.html.
- [27] J. THOREBÄCK, *Digital predistortion the evolution of linearized transmitters for radio base station*, in GHz Centre Workshop: Signal Processing and Amplifiers, 2012.
- [28] J. WOOD, *A Glimpse of the Future*, Microwave Magazine, IEEE, 13 (2012), pp. 60–69.
- [29] D. ZHOU AND V. E. DEBRUNNER, *Novel Adaptive Nonlinear Predistorters Based on the Direct Learning Algorithm*, Signal Processing, IEEE Transactions on, 55 (2007), pp. 120 –133.
- [30] A. ZHU, P. DRAXLER, J. YAN, T. BRAZIL, D. KIMBALL, AND P. ASBECK, *Open-Loop Digital Predistorter for RF Power Amplifiers Using Dynamic Deviation Reduction-Based Volterra Series*, Microwave Theory and Techniques, IEEE Transactions on, 56 (2008), pp. 1524 –1534.
- [31] A. ZHU, J. PEDRO, AND T. BRAZIL, *Dynamic Deviation Reduction-Based Volterra Behavioral Modeling of RF Power Amplifiers*, Microwave Theory and Techniques, IEEE Transactions on, 54 (2006), pp. 4323 –4332.