NTNU

Norwegian University of
Science and Technology

Characterization of Power Amplifiers by means of nonlinear Models with Memory

Øyvind Kinden Lensjø

Master of Science in Communication Technology
Submission date: June 2008
Supervisor: Nils Holte, IET

Norwegian University of Science and Technology
Department of Electronics and Telecommunications

Problem Description

Bandwidth is an increasingly sought after resource in radio communication. In order to use the available spectrum as efficiently as possible, more bandwidth efficient modulation schemes have to be investigated. These schemes will require linear amplifiers, and these amplifiers are a significant component in any radio communication system when considering price and complexity. However, power amplifiers are inherently nonlinear devices, and this causes problems when employing the bandwidth efficient modulation schemes.

Thus it is of great interest to study ways to linearize such amplifiers. The methods in question are based on predistortion using digital signal processing. An important task is an efficient estimation of the amplifier characteristics. Most existing architectures assume that the characteristics can be modeled well by a memoryless nonlinearity. However, this assumption is only valid for relatively small bandwidths. This study will deal with the modeling of power amplifiers using a combination of nonlinear memoryless characteristics and linear filters.

In addition to this, system-level modeling of power amplifiers is considered. These models are used when simulating the behaviour of the complete radio communication system. Such simulation tools are important for baseband system-designers. An accurate model of the power amplifier is very important for the high performance demanded of such simulation tools.

Assignment given: 28. January 2008
Supervisor: Nils Holte, IET

Abstract

Wireless communication systems suffer a great deal from the inherent nonlinear nature of power amplifiers. Effects such as signal distortion and spectral broadening can seriously degrade the performance of both the system at hand and neighboring ones. These problems need to be dealt with, both in the sense of mitigating them and observing the implications caused. For the case of problem mitigation, some sort of linearization of the power amplifier is needed, while simulation tools provide a way of observing the implications. Data from three different power amplifiers are used in the experiments carried out in this study. All three sets use 16-QAM modulation and have a relatively narrow bandwidth of about 190 kHz.

Digital predistortion is a well known technique for linearization of power amplifiers. An important part of this technique is the modeling of the power amplifier. In this study, two different models are investigated. Both models employ cubic splines to approximate the nonlinearity. The first model is memoryless, consisting only of the spline approximation, while the second one also accounts for memory effects by inserting a linear transversal filter subsequent to the spline approximation. The models have been tested and evaluated, proving the model with memory effects to be the superior one. However, this superiority is by no means general.

Radio communication simulation tools need accurate models of power amplifiers to achieve high performance. This area of modeling power amplifiers is called system-level modeling. Two empirical models have been investigated in this study, both applicable as system-level models. Both models exploit physically-based memory effects related to the RF frequency response, self-heating and the biasing network to obtain highly accurate results. The models differ only in the basis functions employed when approximating nonlinearities; the first model employs polynomials while the second one employs cubic splines. The two models have been tested and evaluated, both performing at a very high level. The empirical models presented have many degrees of freedom, and it was discovered that the optimal structure of the model depended on the power amplifier it was set to characterize. Hence, the optimal structures of the models were found not to be general.

The results obtained are evaluated both in time- and frequency-domain. Problems and limitations concerning both the models themselves and the model coefficient estimation procedures are analyzed and discussed along with proposals for further work.

Preface

This master thesis was submitted to the Department of Electronics and Telecommunications at the Norwegian University of Science and Technology (NTNU). The work presented was carried out during the spring of 2008 and is a continuance of the project thesis written in the previous semester. The work has been both interesting and difficult. Challenges have been overcome through hard work and with insightful help from my supervisor Professor Nils Holte. I would like to thank him for all his assistance during this spring.

I would also like to thank my fellow students Christian, Børge, Frode and Lars Petter for interesting discussions, good company and for helping me using LaTeX.

Øyvind Kinden Lensjø
June 2008

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the later years the world has seen a large increase in the number of wireless applications. The development has been remarkable, from cell phones to WLAN to sensor networks, wireless applications are now an important part of everyday life. With new applications entering the market all the time, it is safe to say that the development is still going strong. Of course, every one of these needs some transmission bandwidth in the radio frequency spectrum to be able to operate. With the dramatic increase in the number of entrants in the market, bandwidth has become a much sought after resource. However, the radio frequency spectrum is limited, and as the growth continues, bandwidth efficient modulation schemes are becoming more popular. These schemes require linear power amplifiers (PAs) in order to achieve high performance. However, PAs are inherently nonlinear devices, and as they are indispensable in virtually all radio communication systems, problems arise when employing these modulation schemes.

Thus it is of great interest to study methods that can be used to linearize PAs. Presently proposed solutions are based on predistortion of the signal using digital signal processing. The predistortion operation is performed prior to sending the signal through the PA. A major part of the predistortion technique is the modeling of the amplifier, as it is the first step in designing the predistorter. Most existing architectures assume that the PA characteristic can be modeled well by a memoryless nonlinearity. This assumption is in particular applied in narrowband and low-power applications [1]. However, for applications outside these categories architectures taking into account memory effects have to be considered. It should also be noted that these architectures often outperform the memoryless models also in the narrowband and low-power cases.

The modeling of the PA in the predistortion technique is, as mentioned above, the first step in designing the predistorter. However, this limits the complexity of the PA models as the difficulty of designing the predistorter increases rapidly with more complex PA models. System-level modeling aims exclusively at modeling the PAs as accurately as possible. The models are used when simulating the behaviour of the entire radio communication system. Hence, the difficulty of the predistorter design is no longer a limiting factor as these models are not used in the predistortion design, but as a part of simulation tools. The problem faced is still the inherent nonlinear nature of the PA and the resulting problems when employing bandwidth efficient modulation schemes. System-level modeling of PAs provides the baseband system-designers with a useful tool when designing radio communication systems, as the problems caused by the PA can be discovered and described with high accuracy in simulations. This enables the designers to take those problems into account when designing the systems.

1.1 Objective

The study has two main objectives:

- The predistortion technique provides a way to mitigate the problems caused by the nonlinear nature of PAs when employing bandwidth efficient modulation schemes. The basics of this technique are presented. PA models suited to work well in the predistortion technique are investigated. Both a memoryless model and a model taking into account memory effects are presented, tested and evaluated.
- Employing more complex PA models, we embark on the area of system-level modeling, an important aspect of system simulation tools. System-level modeling of PAs and its objectives are presented briefly. Two models designed for system-level modeling are presented, tested and evaluated.

1.2 Related Work

The area of modeling power amplifiers, both concerning predistortion and system-level simulations, has been a hot topic for research in many years, resulting in many published articles. Given today´s huge interest in wireless applications, the area is still in development. Most modeling approaches presented in the past are based on using Look-Up-Tables, polynomials or piecewise polynomials when modeling the PA characteristic. The employment of splines as basis functions is a relatively new approach, continued in this thesis. The study is mainly a continuance of three different earlier works. They are:

- The project thesis leading up to this master thesis [2]. This project thesis presents the same memoryless model as the one presented here. Hence, much of what is presented on the memoryless model is taken directly from [2].
- [3] presents the theory used to model the memoryless PA characteristic by means of splines.
- [4] presents a physically-based empirical model to be used in system-level modeling. The two system-level models are a continuance of this model.

1.3 Scope

An outline of predistortion is presented along with two models well suited for this technique. As presented above, the modeling of the PA is the first step in designing the corresponding predistorter. However, the actual predistorter design is outside the scope of the study. As for system-level modeling, two models designed for this purpose are presented. They are both designed to work well as a part of a complete radio communication system simulation tool. The actual implementation of such a simulation tool is well beyond the scope of this study, confined to deal with the modeling of the power amplifier.

1.4 Outline

The study is organized as follows:

Chapter 2 presents the background theory of the problems experienced when using nonlinear PAs in radio communication systems. The predistortion technique and the basics of system-level

modeling are described briefly, followed by the presentation of the various models investigated in the study.
Chapter 3 presents how the models were implemented along with problems and limitations concerning this implementation. The data sets used in the modeling are described, as well as the mathematical choices made when employing these data sets in the estimation of the models' coefficients. The system-level models have a lot of selectable parameters. The procedure in which these are chosen is also presented here. The chapter ends with a brief presentation of how the results will be presented later in the study.
Chapter 4 presents the results obtained in the study along with an analysis of these.
Chapter 5 ends the study with a conclusion as well as proposals for further work.

Chapter 2

Background Theory

In order to solve a problem, some basic theory concerning the problem is needed. What is the problem, why do we need to solve it and how do we go about seeking solutions are all important questions in this context. This chapter answers these questions by first presenting the problem and its consequences thoroughly followed by the theory used when attempting to solve it.

2.1 Power Amplifier

Power amplifiers are inherently nonlinear devices and are used in virtually all communication systems [5]. Figure 2.1 depicts a typical PA power characteristic, with input and output power on the horizontal and vertical axis, respectively. The nonlinear behaviour is evident from this

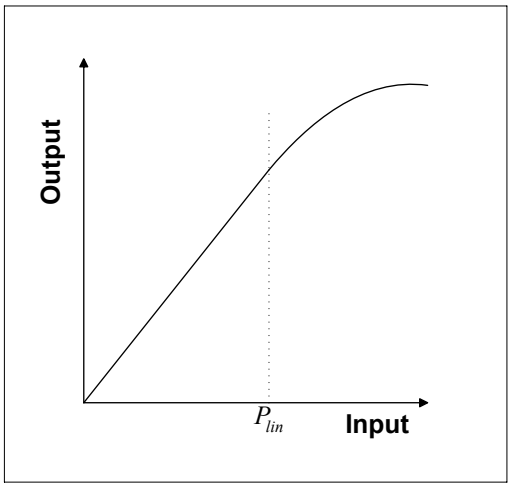


Figure 2.1: PA power characteristic.

figure. Below input power P_{lin} the PA output is a linear function of the PA input, but for larger inputs the output will no longer follow this linear relationship. This nonlinearity can impose large problems in radio communication systems. One of the main reasons for this is found in the recent development in modulation. As more wireless communication systems have entered the market, bandwidth has become a very popular resource, and so, demand for higher spectral efficiency forces these systems to employ non-constant envelope modulation schemes such as QAM[1] [1, 3]. However, when using these kinds of modulation schemes, the nonlinear effects of the PA cause

[1]Quadrature Amplitude Modulation [6].

both a distortion of the signal and an increased out of band spectrum, which leads to a rise in adjacent channel interference [1]. Many radio systems, satellite systems in particular, use one common amplifier for many different carrier frequencies (e.g. when employing FMDA[2]). The nonlinear part of the PA may in this scenario cause a phenomenon known as intermodulation noise, which can cause seriuos performance degradation. For more on this consult [7]. Thus we must find linearization techniques to help us mitigate these problems. A simple model of a PA with its input and output is depicted in figure 2.2. $x(t)$ here represents the input signal to

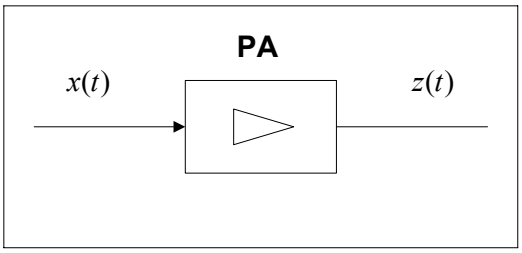


Figure 2.2: Simple PA model.

the PA and $z(t)$ the output signal, both complex valued. A linear PA will have the following relationship between these two signals:

$$z(t) = \begin{cases} cx(t) & |x(t)| \leq |\ x(t)|_{max} \\ cx(t)_{max} & |x(t)| > |x(t)|_{max} \end{cases} \tag{2.1}$$

The complex constant c is here the desired linear gain. Hence, the output signal is a linear function of the input signal up to a maximum amplitude $|x(t)|_{max}$. The power of an analog signal is calculated in the following way [8]:

$$P = \frac{1}{T_0} \int_{T_1}^{T_1+T_0} |x(t)|^2 dt \tag{2.2}$$

where T_0 is the time interval over which the power is calculated. Now, using both (2.1) and (2.2), the following relationship between the input and output power in the linear region is:

$$P_{out} = \frac{1}{T_0} \int_{T_1}^{T_1+T_0} |z(t)|^2 dt = \frac{1}{T_0} \int_{T_1}^{T_1+T_0} |cx(t)|^2 dt = \frac{|c|^2}{T_0} \int_{T_1}^{T_1+T_0} |x(t)|^2 dt = |c|^2 P_{in} \tag{2.3}$$

This calculation shows that (2.1) ensures a linear power relationship up to and including the power of $x(t)_{max}$. Above the input power of $x(t)_{max}$ the output power will be a constant $|c|^2 P_{lin}$, where P_{lin} equals the power of $x(t)_{max}$. The characteristic of this PA is depicted in figure 2.3. One way to obtain this is through digital predistortion, outlined in the next section. This involves some digital processing of the input signal prior to sending it through the PA.

[2]Frequency Division Multiple Access [6, 7].

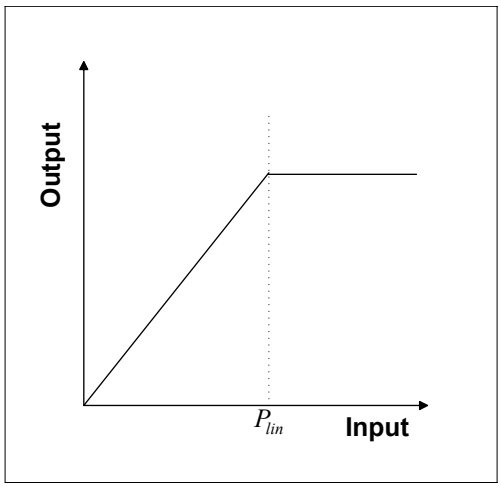


Figure 2.3: Linearized PA power characteristic.

2.2 Predistortion

A system employing digital block based predistortion is depicted in figure 2.4. The predistorter

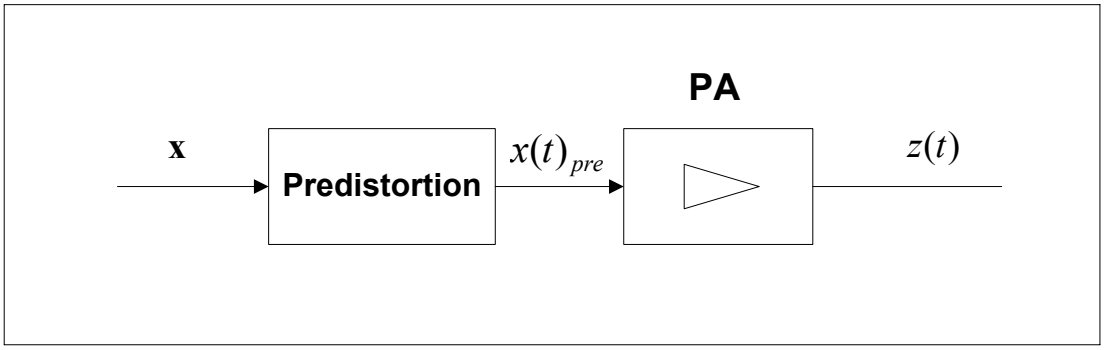


Figure 2.4: A system employing predistortion.

is here a functional block which precedes the PA. The input to the predistorter is digital and the output is analog. The analog output presented in the figure is a simplification of reality[3], but this is irrelevant in this context. If the predistorter is designed properly, the relationship between signals $x(t)_{pre}$ and $z(t)$ will follow equation (2.1), with $x(t)_{pre}$ instead of $x(t)$. Digital predistortion is characterized by robust operation, moderate implementation cost and high accuracy [1], making it a suitable choice for the PA linearization operation.

In order to design and implement a proper predistorter we need a model of the PA. As mentioned in the introduction, most existing architectures assume that the PA characteristic can be modeled well by a memoryless nonlinearity. Figure 2.5 depicts how the predistorter is implemented in this scenario. The same simplification noted above concerning figure 2.4 also applies now. The PA is modeled as $g(\cdot)$. This modeling procedure is the topic of section 2.4. As shown in the figure the predistorion function is now simply the inverse of the PA model. This function can be calculated analytically.

For systems employing non-constant envelope modulation schemes, PA memory effects can no

[3] $\mathbf{x}$ is here the symbol sequence. The block called Predistorter will in reality contain all operations performed on the sequence before it is ready to be transmitted (e.g. pulse shaping).

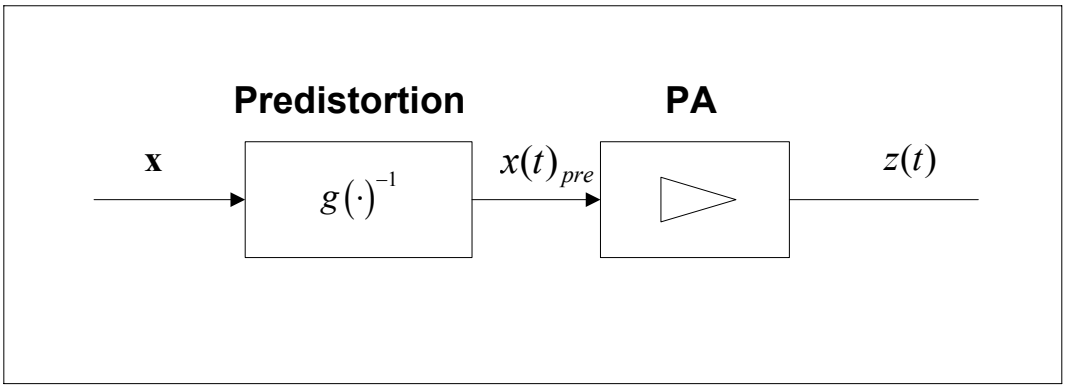


Figure 2.5: Memoryless predistortion.

longer be ignored [1]. This is also the case for wide bandwidth applications such as WCDMA[4] [5]. These memory effects have to be accounted for in the PA model. When working with digital block based predistortion, introducing one or two filters in the model is a mathematical convenient technique. This is mainly because the design and implementation of the corresponding predistorter (inverting the model) can be calculated analytically with limited complexity. The introduction of the filters is depicted in figure 2.6a, along with the corresponding predistorter structure in figure 2.6b. All signals are here digital, representing sampled versions of the analog input and output. As shown in the figure the memoryless nonlinearity $g(\cdot)$ is still a part of the

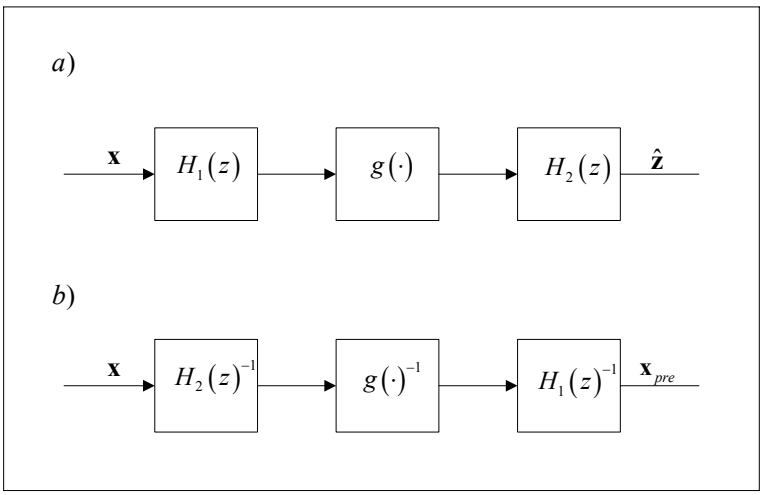


Figure 2.6: Predistortion with memory.

model. The filters $H_1(z)$ and $H_2(z)$ models the memory effects. With this kind of structure, three models exist. They are:

- The Wiener model. In this case $H_2(z) = 1$[5]. Hence the PA model consists of a filter $H_1(z)$ followed by a memoryless nonlinearity $g(\cdot)$.
- The Hammerstein model. In this case $H_1(z) = 1$. Hence the model consists of a memoryless

[4] Wideband Code Division Multiple Access.

[5] $H_1(z)$ and $H_2(z)$ are the z-transforms [9] of the two filters. When the z-transform equals 1, the filter is only a delta-pulse, in other words no filter at all; the output equals the input. In this situation the z-transform of the inverse filters in the predistorter will also be 1.

nonlinearity $g(\cdot)$ followed by a filter $H_2(z)$.

- The Wiener-Hammerstein model. In this case both the filters will be part of the model, which now consists of a filter $H_1(z)$ followed by a memoryless nonlinearity $g(\cdot)$ followed by a filter $H_2(z)$.

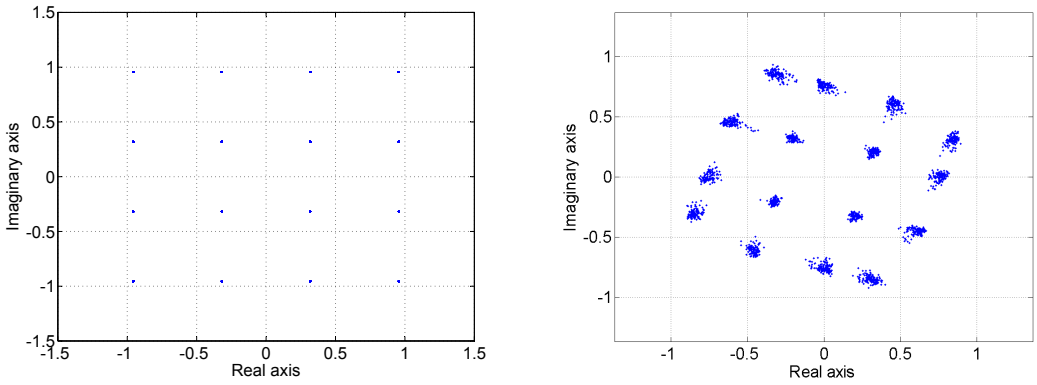
As described above, in order to design and implement a proper predistorter, the PA has to be modeled mathematically. This modeling is one of the main topics of this study, while the inversion of the models will not be treated. Four PA models have been presented in this section, but only two of them will be investigated; the memoryless model and the Hammerstein model. These two are the most convenient to solve in terms of mathematical complexity. These modeling approaches will be described thoroughly in sections 2.4 and 2.5.

The models presented in this section are all well suited for predistortion. The predistortion function can be calculated analytically once the PA model is estimated. However, the complexity of the inverse operation (finding the predistortion function) constrains the PA model. Thus, when not taking into account the inverse operation and only dealing with the modeling of the PA, more complex models than the ones presented here can be employed. This is the topic of the next section, where system-level modeling of PAs is presented.

2.3 System-Level Modeling

Section 2.1 presented the problems faced when employing nonlinear PAs in radio communication systems. The digital block based predistortion technique outlined in section 2.2 is one way of reducing these problems. However, with this approach, the PA model is limited in complexity by the need to perform the inverse operation. The inverse of the models presented in section 2.2 can, as described, be calculated analytically. Now it is time to turn away from the predistortion technique and have a look at what is called system-level modeling. The aim of system-level modeling is simply to model the PA in the best possible way. This of course leads to a situation where the complexity of the PA model is no longer limited by the difficulty of the inverse operation, often resulting in better fitted models. As mentioned in [4], an efficient way to accurately predict the system-level behavior of PAs provides a useful tool for baseband system-designers to take into account those nonlinearities in their simulations. In other words, a proper PA model enables baseband system-designers to better predict the behavior of the system, as the non-linear effects of the PA is accounted for by the model. As also stated in [4], this will result in a more accurate and reliable simulation tool to evaluate the performance of the whole communication system. Figure 2.7 depicts a simple example of this kind of simulation tool. In this example a 16-QAM modulated symbol sequence has been transmitted through an ideal noiseless channel and then demodulated to get the received symbol sequence. With this scenario (ideal channel and no noise) the received sequence equals the transmitted sequence. Figure 2.7a shows the demodulated signals when using a linear ideal PA and figure 2.7b when using a simple model of a real PA[6]. The nonlinear effects of the PA are clear, a distortion both in phase and amplitude is observed. In this kind of simulation it is clear that the accuracy of the PA model is of great importance. The more accurate PA model, the more reliable simulation tool. Thus, more complex models are employed in this context than what was the case in section 2.2 dealing with predistortion. Section 2.6 presents two models for this purpose.

[6]The PA model is a memoryless spline model. This kind of model is presented in section 2.4. The figures are taken from [10].



(a) Linear PA. No distortion.

(b) Real PA model. Amplitude and phase distortion.

Figure 2.7: Comparison of a distorted and a nondistorted 16-QAM signal constellation.

2.4 The Memoryless Model

Most of this theory is taken from [3]. It is repeated here as it is crucial for the understanding of the memoryless model, and is also an important part of the models described in the next two sections.

The PA characteristic is assumed to be a memoryless nonlinearity, $g(\cdot)$. Figure 2.8 depicts how the PA is modeled with a) showing how the data samples were obtained and b) showing the model described here. A model with no memory implies that every output sample z_i depends

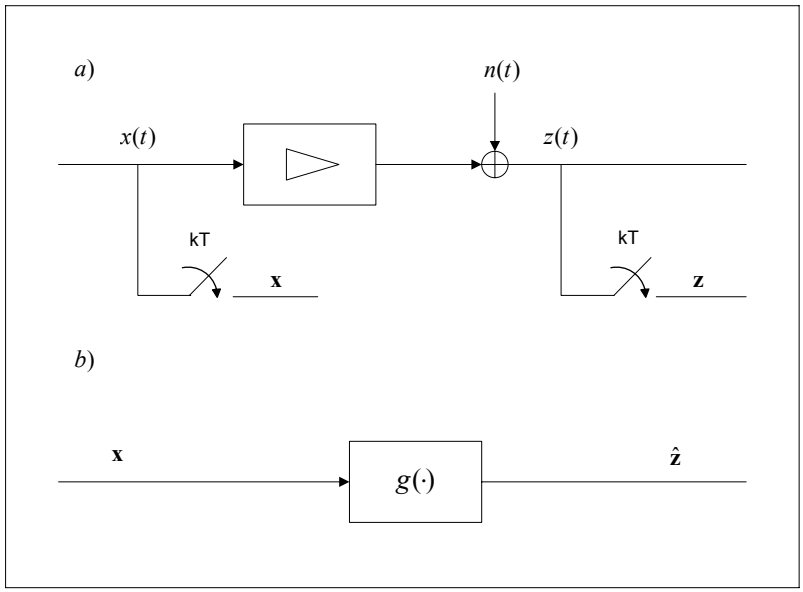


Figure 2.8: The memoryless model.

only on the corresponding input sample x_i;

$$z_i = g(x_i) + n_i \tag{2.4}$$

n_i represents the unavoidable measurement noise. This nonlinearity will be modeled using cubic splines. Splines are piecewise continuous polynomial functions. A cubic spline is a cubic

function in each interval (as described later, the input range is divided into intervals of equal length). The spline approximation is continuous at the borders between intervals, as are both its first and second derivatives. Hence, continuity will automatically be ensured by using splines. This continuity leads to a smooth curve, which is in close correspondence with a typical PA characteristic. For more on splines in general consult [11].

The estimation of the complex characteristic $g(\cdot)$ is performed using a block of PA input and output samples. These samples were obtained as described in section 3.3. Hence, $\mathbf{x}$ represents the input samples and $\mathbf{z}$ represents the output samples. Both these vectors are normalized, meaning:

$$|x_i|, |z_i| \leq 1 \qquad \forall i$$

We want to find a least square estimate of the complex PA characteristic $\hat{g}(\cdot)$ by minimizing

$$J = \sum_{i=1}^{N} w_i |z_i - \hat{z}_i|^2 \tag{2.5}$$

where

$$\hat{z}_i = \hat{g}(x_i) \tag{2.6}$$

and w_i is the appropriate weighting factor described in section 3.3.1. The nonlinear characteristic $\hat{g}(\cdot)$ of the PA is independent of the phase of the input signal, so for simplicity x now means $|x|$. B-splines are used as basis functions, which mean that each basis function is nonzero over a minimum number of intervals. This ensures low computational complexity. Using cubic splines now means that each basis function covers four intervals. The input interval is divided into K intervals of equal length, and the borders between the intervals are denoted knots with values $k_1, \ldots, k_{K-1}$. A general estimate based on B-splines may now be expressed by:

$$\hat{g}(x) = \sum_{j=K_0}^{K-1} a_j B_j(x) \tag{2.7}$$

B_j is here the B-spline starting at knot number j, and a_j is the corresponding coefficient.
$K_0 = -3$ since we are using cubic splines. The B-spline with index j is simply a translation of $B_0(x)$:

$$B_j = B_0(x - k_j) \tag{2.8}$$

Each input sample x is decomposed into an interval number k_x and the corresponding normalized fraction of the interval χ given by:

$$k_x = \lfloor K \cdot x \rfloor \tag{2.9}$$

$$\chi = K \cdot x - k_x \tag{2.10}$$

The $B_0(x)$ function is given by:

$$B_0(x) = \begin{cases} \frac{1}{4}\chi^3 & x \in [k_0, k_1\rangle \\ -\frac{3}{4}\chi^3 + \frac{3}{4}\chi^2 + \frac{3}{4}\chi + \frac{1}{4} & x \in [k_1, k_2\rangle \\ \frac{3}{4}\chi^3 - \frac{3}{2}\chi^2 + 1 & x \in [k_2, k_3\rangle \\ -\frac{1}{4}\chi^3 + \frac{3}{4}\chi^2 - \frac{3}{4}\chi + \frac{1}{4} & x \in [k_3, k_4] \end{cases} \tag{2.11}$$

When approximating the characteristic of a PA AM/AM[7] characteristic, there will usually be a constraint that the characteristic start at the origin, and that it has maximum smoothness at $x = 0$. For a real characteristic this may be expressed as:

$$\hat{g}(0) = 0 \qquad \hat{g}''(0) = 0 \tag{2.12}$$

The following modification of the basis functions fulfills these constraints: B_{-3} and B_{-2} has to be deleted and B_{-1} must be modified as follows:

$$B_{-1}(x) = \begin{cases} -\frac{1}{2}\chi^3 + \frac{3}{2}\chi & x \in [k_0, k_1\rangle \\ \frac{3}{4}\chi^3 - \frac{3}{2}\chi^2 + 1 & x \in [k_1, k_2\rangle \\ -\frac{1}{4}\chi^3 + \frac{3}{4}\chi^2 - \frac{3}{4}\chi + \frac{1}{4} & x \in [k_2, k_3] \end{cases} \tag{2.13}$$

The rest of the basis functions will follow (2.8) and (2.11). By using this set of basis functions and complex coefficients, the constraints will be satisfied both for the real and the imaginary part of the characteristic, and hence also for the approximated AM/AM characteristic. No constraints are assumed for the AM/PM[8] characteristic. However, the approximation procedure for the complex case implicitly causes the derivative of the AM/PM characteristic to be zero at $x = 0$, but this is unimportant because the phase has no significance in this region[9].

We now switch to matrix notation, in order to simplify mathematics. (2.7) is now[10]:

$$\hat{g}(x) = \mathbf{b}^T(x)\mathbf{a} \tag{2.14}$$

where

$$\mathbf{a}^T = [a_{K_0} \quad a_{K_0+1} \ \ldots \ a_{K-1}] \tag{2.15}$$

$$\mathbf{b}^T(x) = [B_{K_0}(x) \quad B_{K_0+1}(x) \ \ldots \ B_{K-1}(x)] \tag{2.16}$$

The vector of the estimated output based on the observations is now:

$$\hat{\mathbf{g}}^H = [\hat{g}^*(x_1) \quad \hat{g}^*(x_2) \ \ldots \ \hat{g}^*(x_N)] = \mathbf{a}^H\mathbf{V}^H \tag{2.17}$$

where

$$\mathbf{V}^H = [\mathbf{b}(x_1) \quad \mathbf{b}(x_2) \ \ldots \ \mathbf{b}(x_N)] \tag{2.18}$$

This leads to the minimization criteria:

$$J = (\mathbf{z}^H - \mathbf{a}^H\mathbf{V}^H)\mathbf{W}(\mathbf{z} - \mathbf{V}\mathbf{a}) \tag{2.19}$$

where $\mathbf{W}$ is a diagonal matrix containing the weighting factors. The coefficient vector $\hat{\mathbf{a}}$ given by a least square solution is now [12]:

$$\hat{\mathbf{a}} = (\mathbf{V}^H\mathbf{W}\mathbf{V})^{-1}\mathbf{W}\mathbf{V}^H\mathbf{z} \tag{2.20}$$

where $\mathbf{z}$ is the vector of output samples after compensation for the phase of the input signal:

$$\mathbf{z}^T = \left[z_1 \cdot e^{-j(\angle x_1)} \quad z_2 \cdot e^{-j(\angle x_2)} \ \ldots \ z_N \cdot e^{-j(\angle x_N)}\right] \tag{2.21}$$

The memoryless complex characteristic is now given by (2.14) and the estimate of the coefficient vector is defined by (2.20).

[7] Amplitude/Amplitude.
[8] Amplitude/Phase.
[9] See section 4.1 for an explanation of this.
[10] The superscripts T, H and $*$ denotes transpose, Hermitian transpose and complex conjugate, respectively.

2.5 The Hammerstein Model

As described in the previous section; in the memoryless model each output sample depends only on the corresponding input sample. In practice this does not reflect how PAs work in real life, there will always be some sort of memory effects involved. The Hammerstein model takes this into account by inserting a filter subsequent to the modeled memoryless nonlinearity. Thus, the output samples are no longer dependent only on the corresponding input samples, but on multiple input samples. The main idea behind the Hammerstein model is shown in figure 2.9. Figure 2.9a

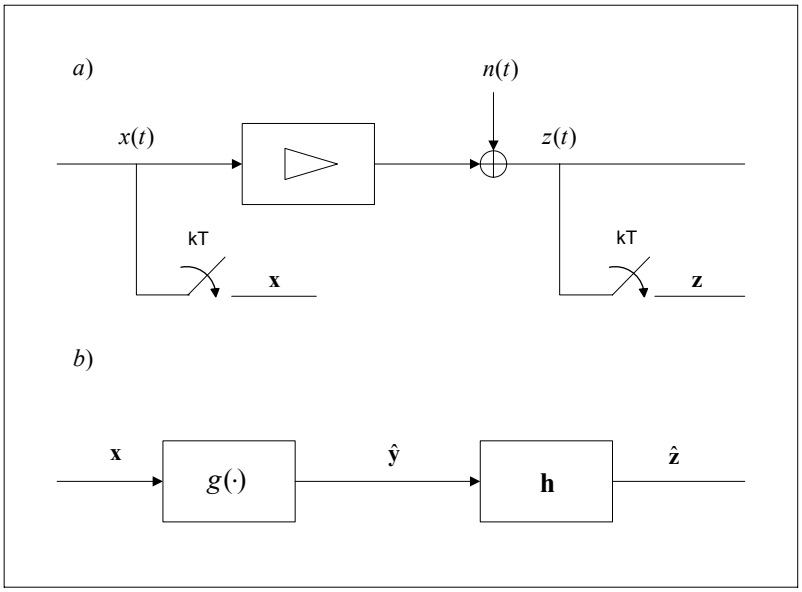


Figure 2.9: The hammerstein model.

shows how the data samples were obtained. $n(t)$ represents the measurement noise in the model. The Hammerstein model is depicted in figure 2.9b. $g(\cdot)$ represents a memoryless nonlinearity, in full correspondence with the one described in the previous section. $\mathbf{h}$ is a vector of filter coefficients, describing the impulse response if the discrete filter $h(n)$. This filter incorporates the memory in the model. The filter described here is a linear transversal filter with coefficients h_i, where $i \in [-L, L]$. The filter length is thus 2L+1. The output samples from the model are now:

$$\hat{z}_i = \sum_{m=-L}^{L} h_m \cdot g(x_{i-m}) \tag{2.22}$$

Both the nonlinearity and the filter have complex valued coefficients.

The minimization criterion is now the weighted mean squared error between the observed and the modeled values:

$$J = \frac{1}{N} \sum_{i=1}^{N} w_i |\hat{z}_i - z_i|^2 \tag{2.23}$$

Hence, we want to find the spline coefficients $\mathbf{a}$ and the filter coefficients $\mathbf{h}$ which minimize J. However, this leads to a nonlinear minimization problem and is thus not easy to solve. In [2] an attempt was made to circumvent this problem by transforming the minimization procedure into a linear problem. However, this procedure did not lead to a general solution. In this study the problem is solved iteratively, using the Optimization Toolbox in MATLAB. The specific function

chosen was *lsqcurvefit*[11] with a user-supplied Jacobian matrix. This function is defined to solve nonlinear curve-fitting problems in a least-squares sense. Thus, the function finds the coefficients $\mathbf{c}$ which minimize the equation:

$$\frac{1}{2}\|F(\mathbf{c},\mathbf{xdata})-\mathbf{ydata}\|_2^2=\frac{1}{2}\sum_{i=1}^{N}|F(\mathbf{c},xdata_i)-ydata_i|^2 \tag{2.24}$$

where $\mathbf{c}$, **xdata** and **ydata** are vectors containing the model coefficients, the input data and the output data, respectively. **xdata** and **ydata** both have length N, and $F(\mathbf{c},\mathbf{xdata})$ is a vector-valued function. The algorithm used to solve the problem is a subspace trust-region method based on the interior-reflective Newton method, described in [13] and [14]. In the previous section the components of the observed input vector $\mathbf{x}$ was treated as the absolute value of the input samples. This is not the case in this section, $\mathbf{x}$ now has complex valued components. The vectors in (2.24) are now:

$$\mathbf{c}=[\mathbf{a},\mathbf{h}]=[a_{K_0}\quad a_{K_0+1}\ \ldots\ a_{K-1}\quad h_{-L}\ \ldots\ h_0\ \ldots\ h_L] \tag{2.25}$$

$$\mathbf{xdata}=\mathbf{x}=[x_1\quad x_2\ \ldots\ x_N] \tag{2.26}$$

$$\mathbf{ydata}=\mathbf{z}=[z_1\quad z_2\ \ldots\ z_N] \tag{2.27}$$

The function $F(\mathbf{c},\mathbf{x})$ is the output of the Hammerstein model:

$$\hat{\mathbf{z}}=F(\mathbf{c},\mathbf{x})=[F(\mathbf{c},x_1)\quad F(\mathbf{c},x_2)\ \ldots\ F(\mathbf{c},x_N)] \tag{2.28}$$

As described in the previous section, and also given by (2.23), weighting of the data is desirable. Thus, we have to multiply both (2.27) and (2.28) by $\sqrt{\mathbf{W}}$, where $\mathbf{W}$ is a diagonal matrix containing the weighting factors. All factors in $\mathbf{W}$ are real valued. Hence (2.24) is now:

$$\frac{1}{2}\|\sqrt{\mathbf{W}}F(\mathbf{c},\mathbf{x})-\sqrt{\mathbf{W}}\mathbf{z}\|_2^2=\frac{1}{2}\sum_{i=1}^{N}w_i|\hat{z}_i-z_i|^2 \tag{2.29}$$

When comparing (2.29) with (2.23) these differ only in the constant preceding the sum ($\frac{1}{2}$ vs. $\frac{1}{N}$). However, since this is the function to be minimized this will have no influence on the minimization procedure. The output from the Hammerstein model is given in (2.22). However, the $g(x_i)$ displayed in this formula is not the same as the one used in the previous section. Because the nonlinear characteristic is independent of the phase of the input, the phase of each sample is inserted after the memoryless nonlinearity[12]. Hence we now write:

$$g(x_i)=g(|x_i|)\cdot e^{j\angle x_i}=\sum_{j=K_0}^{K-1}a_jB_j(|x_i|)\cdot e^{j\angle x_i} \tag{2.30}$$

Now, using both (2.22) and (2.30), the output from the Hammerstein model is:

$$\hat{z}_i=\sum_{m=-L}^{L}\sum_{j=K_0}^{K-1}h_ma_jB_j(|x_{i-m}|)\cdot e^{j\angle x_{i-m}} \tag{2.31}$$

[11] All info about this function can be found at www.mathworks.com. The large-scale type of the function was used in this study.

[12] This is not depicted in figure 2.9b, but this has no significance since the input is now complex and $g(\cdot)$ will only depend on the magnitude.

As described above, the problem is solved iteratively. This means that the expression presented in (2.31) will have to be calculated for every sample in every iteration. If we set the calculated coefficient vector at time t to $\mathbf{c}^{(t)}$, the actual value used in the minimization procedure at time $t+1$ will be

$$\hat{z_i}^{(t+1)} = \sum_{m=-L}^{L} \sum_{j=K_0}^{K-1} h_m^{(t)} a_j^{(t)} B_j(|x_{i-m}|) \cdot e^{j\angle x_{i-m}} \tag{2.32}$$

The Jacobian matrix is also needed to solve the problem as efficiently as possible. When using the *lsqcurvefit* function, the Jacobian matrix is calculated by deriving all the modeled values with respect to all the parameters in the following fashion:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \sqrt{w_1} F(\mathbf{c},x_1)}{\partial c_1} & \cdots & \frac{\partial \sqrt{w_N} F(\mathbf{c},x_N)}{\partial c_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial \sqrt{w_1} F(\mathbf{c},x_1)}{\partial c_m} & \cdots & \frac{\partial \sqrt{w_N} F(\mathbf{c},x_N)}{\partial c_m} \end{bmatrix} \tag{2.33}$$

Of course, the Jacobian matrix also has to be recalculated in every iteration. Using the same time notation as above, and completing the derivations we have the following Jacobian matrix at time $t+1$:

$$\mathbf{J}^{(t+1)} = \begin{bmatrix} J_{spline}(x_1)_{a_{K_0}}^{(t)} & \cdots & J_{spline}(x_N)_{a_{K_0}}^{(t)} \\ \vdots & \ddots & \vdots \\ J_{spline}(x_1)_{a_{K-1}}^{(t)} & \cdots & J_{spline}(x_N)_{a_{K-1}}^{(t)} \\ J_{filter}(x_1)_{-L}^{(t)} & \cdots & J_{filter}(x_N)_{-L}^{(t)} \\ \vdots & \ddots & \vdots \\ J_{filter}(x_1)_{L}^{(t)} & \cdots & J_{filter}(x_N)_{L}^{(t)} \end{bmatrix} \tag{2.34}$$

where

$$J_{spline}(x_i)_{a_j}^{(t)} = \sum_{m=-L}^{L} \sqrt{w_i} h_m^{(t)} B_j(|x_{i-m}|) \cdot e^{j\angle x_{i-m}} \tag{2.35}$$

$$J_{filter}(x_i)_{m}^{(t)} = \sum_{j=K_0}^{K-1} \sqrt{w_i} a_j^{(t)} B_j(|x_{i-m}|) \cdot e^{j\angle x_{i-m}} \tag{2.36}$$

When solving the problem in an iterative manner such as this, one must provide the function with an initial vector $\mathbf{c_0}$. However, if the minimization object, in this case (2.29), is not strictly convex (which it is assumed not to be) the choice of the initial vector may have crucial influence on the final result as it may lead to local solutions[13]. The choice made in this study was to use the results from the memoryless model as the initial $\mathbf{a}$. The initial filter has a centertap set to 1 and small complex numbers (in the order of 2 % and lower) for the rest of the taps. The complex numbers decrease with increasing distance from the centertap[14]. Thus, the initial vector is:

$$\mathbf{c}_0 = \begin{bmatrix} a_{K_0}^{ml} & \dots & a_{K-1}^{ml} & c_{-L}^{(init)} & \dots & c_{-1}^{(init)} & 1 & c_1^{(init)} & \dots & c_L^{(init)} \end{bmatrix} \tag{2.37}$$

where the superscript ml refers to the results obtained when estimating the memoryless model.

[13] If the minimization object is not strictly convex, one or more local minimums, in addition to the global minimum, exists [15]. Depending on the choice of the initial vector one can not be sure that *lsqcurvefit* converges to the global minimum and not a local minimum.

[14] The numbers are chosen as small complex numbers (and not zero) to avoid a situation where the iterative calculation starts at a local minimum and is "trapped" in this minimum. These numbers decrease as the closest samples are most likely to contribute to the memory effects.

2.6 The Empirical Model

This section presents the two models mentioned in section 2.3 on system-level modeling. The models are a continuance of work presented by N. Safari in [4]. Some of the theory presented in this section is taken directly from [4] as it is necessary for the understanding of the models. The main idea is to exploit some physical knowledge of the PA to build a more accurate model. [4] also aims at reducing the complexity of the model by reducing the number of parameters as much as possible. This is not considered here, the only aim is to model the PA as accurately as possible.

The model proposed is a physically-based model including thermal and biasing effects to establish a discrete-time input/output empirical model of power amplifiers for both narrowband and wideband applications. Figure 2.10 depicts this model. As shown the model includes three branches. The top branch consists of a Hammerstein model (see section 2.5). This will model the PA memoryless nonlinearity and the RF[15] frequency response. The output from this branch, $\widetilde{\mathbf{u}}$, is then multiplied by $\widetilde{\mathbf{v}}$ which contains the thermal and biasing effects. This product gives the model output $\widetilde{\mathbf{z}}$.

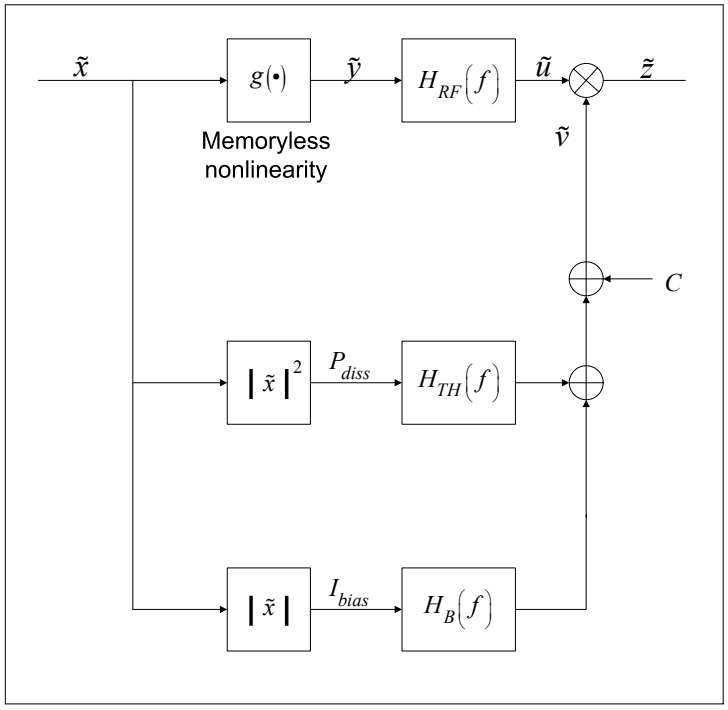


Figure 2.10: The empirical general model.

The thermal effects of the PA is modeled based on filtering of the dissipated power giving the device´s junction temperature. The variations in the device´s junction temperature modulates the transconductance of the transistor and hence the gain of the PA. The dissipated power is assumed to be equal to the instantaneous input power. The thermal filter $H_{TH}(f)$ is in this model a recursive filter[16].

A simple model is considered for the memory effects caused by the bias network. The base-band current of the PA is assumed to be linearly proportional to the envelope of the input signal.

[15] Radio Frequency.
[16] In [4] this filter is modeled as a causal low-pass filter.

The variation of the input envelope is then filtered by $H_B(f)$ and then modulates the gain of the PA.

In the following the two (slightly) different models will be presented, both based on the general model shown in figure 2.10.

2.6.1 The Empirical Polynomial Model

The empirical polynomial model is pretty much the same as the model presented in [4], though with some extensions. The basis functions used in the memoryless nonlinearity are polynomials. This means that the output passband signal from $g(\cdot)$, in discrete-time, can be expressed as:

$$\widetilde{y}_i = \sum_{k=1}^{K} \widetilde{d}_k \widetilde{x}_i^k \tag{2.38}$$

where K is the maximum order of the polynomial that approximates the PA memoryless nonlinearity. All signals in the model are now passband signals, thus are $\widetilde{d}_k$ real valued coefficients. $\widetilde{y}_i$ is filtered by (the discrete version of) $H_{RF}(f)$ representing the RF frequency response of the PA. By assuming a FIR[17] filter the output from the top branch may now by expressed by:

$$\widetilde{u}_i = \sum_{q=-Q_1}^{Q_2} \widetilde{h}_q^{(rf)} \widetilde{y}_{i-q} = \sum_{q=-Q_1}^{Q_2} \sum_{k=1}^{K} \widetilde{a}_{q,k} \widetilde{x}_{i-q}^k \tag{2.39}$$

where $\widetilde{a}_{q,k} = \widetilde{h}_q^{(rf)} \cdot \widetilde{d}_k$ are the coefficients of the cascade model in the top branch. The filter is a non-causal filter with length $Q_1 + Q_2 + 1$[18]. The coefficients $\widetilde{a}_{q,k}$ are now considered as a (2-D) array over q and k. Hence, the desired linear relationship between the output of the model and the coefficients can be taken advantage of. The coefficients can now be simply identified using any least square algorithm. At this point it is important to emphasize that the top branch is no longer a proper Hammerstein model, as it is impossible to separate the coefficients of the memoryless nonlinearity ($\widetilde{d}_k$) and the filter ($\widetilde{h}_q^{rf}$).

The next step is to obtain the baseband representation of the top branch. Once this is done the coefficients may be identified using a time-series analysis of the baseband signal.

The real input signal, $\widetilde{x}_i$, can be expressed in terms of its complex baseband representation, x_i, as:

$$\widetilde{x}_i = \mathbf{Re}\{e^{j\omega_0 i} x_i\} = \frac{1}{2}[e^{j\omega_0 i} x_i + e^{-j\omega_0 i} x_i^*] \tag{2.40}$$

D. R. Morgan *et. al.* showed in [16] that for signals whose bandwidth is small compared to the center frequency ($f_c >> BW$), $\widetilde{x}_i^k = \frac{1}{2^k}[e^{j\omega_0 i} x_i + e^{-j\omega_0 i} x_i^*]$ can generate intermodulation products close to the transmit band only if k is an odd number and the number of x_i terms differs from x_i^* terms by one. Using this concept and considering only the combinations where the number of x_i terms is larger than the number of x_i^* terms by exactly one, we may replace $\widetilde{x}_i^k$ by $x_i \cdot x_i^{(\frac{k-1}{2})} \cdot (x_i^*)^{(\frac{k-1}{2})} = x_i \cdot |x_i|^{(k-1)}$ for odd k.

Based on the discussion made above, the baseband input/output relationship for the top branch of the empirical polynomial model is given by:

$$u_i = \sum_{q=-Q_1}^{Q_2} \sum_{\substack{k=1 \\ k:odd}}^{K} a_{q,k} x_{i-q} |x_{i-q}|^{k-1} \tag{2.41}$$

[17] Finite Impulse Response.

[18] In [4] this filter is causal.

In equation (2.41) x_i and u_i are baseband complex input and output signals, and $a_{q,k}$ are complex valued coefficients.

Now we consider the second and third branch of the model. As figure 2.10 shows the power and the envelope is passed to the branch modeling the thermal effects and the branch modeling the biasing effects, respectively. The specific structure of these two branches is depicted in figure 2.11, showing the complete empirical polynomial model. As shown, the filter in the second branch is a recursive filter. Hence, the baseband output signal is:

$$v_i^{(th)} = \sum_{j=0}^{j<i} |x_{i-j}|^2 \rho^j \tag{2.42}$$

where ρ is a real number following:

$$\rho < 1 \tag{2.43}$$

The baseband output from the third branch modeling the biasing effects is:

$$v_i^{(bi)} = \sum_{m=-M_1}^{M_2} h_m^{(bi)} |x_{i-m}| \tag{2.44}$$

Hence, (the discrete version of) $H_B(f)$ is a non-causal FIR filter with length $M_1 + M_2 + 1$[19]. The constant C in figure 2.10 is now set to 1. It is inserted in the model to preserve the linear terms in the first branch after the multiplication. The complete output signal from the lower branches is now:

$$v_i = \sum_{j=0}^{j<i} |x_{i-j}|^2 \rho^j + \sum_{m=-M_1}^{M_2} h_m^{(bi)} |x_{i-m}| + 1 \tag{2.45}$$

in full correspondence with figure 2.11.

Finally, multiplying the two sets of components obtained in equation (2.41) and (2.45), we may represent the discrete time input/output of the PA baseband model as follows:

$$\begin{aligned} z_i = & \sum_{q=-Q_1}^{Q_2} \sum_{\substack{k=1 \\ k:odd}}^{K} a_{q,k} x_{i-q} |x_{i-q}|^{(k-1)} + \\ & \sum_{q=-Q_1}^{Q_2} \sum_{\substack{k=1 \\ k:odd}}^{K} a_{q,k}^{(th)} x_{i-q} |x_{i-q}|^{(k-1)} v_i^{(th)} + \\ & \sum_{q=-Q_1}^{Q_2} \sum_{\substack{k=1 \\ k:odd}}^{K} \sum_{m=-M_1}^{M_2} a_{q,k,m}^{(bi)} x_{i-q} |x_{i-q}|^{(k-1)} |x_{i-m}| \end{aligned} \tag{2.46}$$

The reason why $v_i^{(th)}$ is not written out in full detail in equation (2.46) is that is will be calculated in advance. Hence the only difference between branch one and two is that the basis functions from the top branch are modified when multiplied with $v_i^{(th)}$. The coefficients $a_{q,k}^{(th)}$ are, as $a_{q,k}$, considered as a (2-D) array over q and k. The coefficients $a_{q,k,m}^{(bi)}$ are considered as a (3-D) array over q, k and m, with $a_{q,k,m}^{(bi)} = h_q^{(rf)} \cdot d_k \cdot h_m^{(bi)}$. Of course, $h_q^{(rf)}$ and d_k are the baseband equivalents of the passband model coefficients $\widetilde{h}_q^{(rf)}$ and $\widetilde{d}_k$ treated above. As stated above in

[19] In [4] this filter is causal.

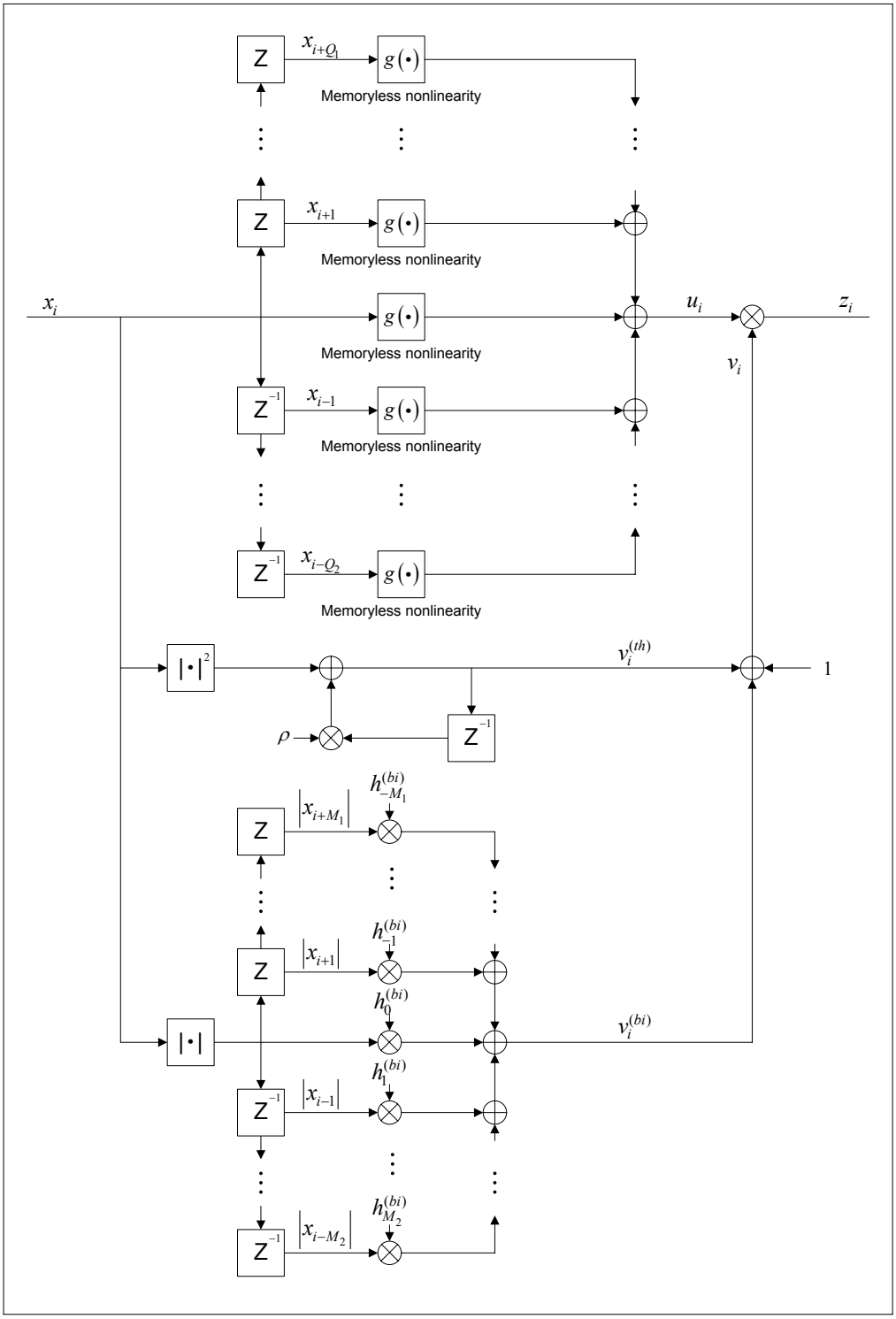


Figure 2.11: The empirical polynomial model.

the discussion of the top branch, when treating the coefficients as arrays in this way, the linear relationship between the output and the model coefficients makes the problem solvable with any least square algorithm. It is important to note that it is this operation that transforms the model into an empirical model (from a purely physically-based one) as it is no longer possible to draw conclusions on the filter coefficients presented in figure 2.10.

Not knowing the time aspect of the memory effects in question, capturing of the long-term memory effects is obtained by introducing a sparse delay-tap structure, i.e. only some of the filter coefficients $h_q^{(rf)}$ and $h_m^{(bi)}$ are nonzero and the rest is set to zero. Using uniform sparse delay-tap, the complete empirical polynomial model is now:

$$\begin{aligned} z_i = & \sum_{q=-Q_1}^{Q_2} \sum_{\substack{k=1 \\ k:odd}}^{K} a_{q,k} x_{i-q'} |x_{i-q'}|^{(k-1)} + \\ & \sum_{q=-Q_1}^{Q_2} \sum_{\substack{k=1 \\ k:odd}}^{K} a_{q,k}^{(th)} x_{i-q'} |x_{i-q'}|^{(k-1)} v_i^{(th)} + \\ & \sum_{q=-Q_1}^{Q_2} \sum_{\substack{k=1 \\ k:odd}}^{K} \sum_{m=-M_1}^{M_2} a_{q,k,m}^{(bi)} x_{i-q'} |x_{i-q'}|^{(k-1)} |x_{i-m'}| \end{aligned} \tag{2.47}$$

where the delay terms $q^{'}$ and $m^{'}$ are given by:

$$q^{'} = d^{(rf)} \cdot q \tag{2.48}$$

$$m^{'} = d^{(bi)} \cdot m \tag{2.49}$$

The parameters $d^{(rf)}$ and $d^{(bi)}$, selecting the new delay structure of the model, are integer numbers.

Equation (2.47) describes the relationship between the input and the output samples in the empirical polynomial model. As seen, the output is a linear function of the model coefficients, and as described above, this implies that the minimization problem is linear and can thus be solved by any least square algorithm. We now change to matrix notation in order to simplify mathematics. Equation (2.47) is now:

$$\mathbf{z} = \mathbf{X}\mathbf{a} \tag{2.50}$$

where $\mathbf{a}$ is a vector containing all the coefficients:

$$\mathbf{a} = \begin{bmatrix} a_{-Q_1,1} & \dots & a_{Q_2,K} & a_{-Q_1,1}^{(th)} & \dots & a_{Q_2,K}^{(th)} & a_{-Q_1,1,-M_1}^{(bi)} & \dots & a_{Q_2,K,M_2}^{(bi)} \end{bmatrix}^T \tag{2.51}$$

and

$$\mathbf{X} = \begin{bmatrix} \mathbf{\Gamma}(x_1) & \mathbf{\Gamma}(x_2) & \dots & \mathbf{\Gamma}(x_i) & \dots & \mathbf{\Gamma}(x_N) \end{bmatrix}^T \tag{2.52}$$

where

$$\begin{aligned} \mathbf{\Gamma}(x_i) = [& \gamma_{-Q_1,1}^{(rf)}(x_i) \ \dots \ \gamma_{Q_2,K}^{(rf)}(x_i) \quad \gamma_{-Q_1,1}^{(th)}(x_i) \ \dots \\ & \gamma_{Q_2,K}^{(th)}(x_i) \quad \gamma_{-Q_1,1,-M_1}^{(bi)}(x_i) \ \dots \ \gamma_{Q_2,K,M_2}^{(bi)}(x_i)]^T, \end{aligned} \tag{2.53}$$

$$\gamma_{q,k}^{(rf)}(x_i) = x_{i-q'} |x_{i-q'}|^{(k-1)}, \tag{2.54}$$

$$\gamma_{q,k}^{(th)}(x_i) = x_{i-q'} |x_{i-q'}|^{(k-1)} v_i^{(th)} \quad \Big(= x_{i-q'} |x_{i-q'}|^{(k-1)} \sum_{j=0}^{j<i} |x_{i-j}|^2 \rho^j \Big), \tag{2.55}$$

$$\gamma_{q,k,m}^{(bi)}(x_i) = x_{i-q'}|x_{i-q'}|^{(k-1)}|x_{i-m'}|. \tag{2.56}$$

As in the two previous sections we now want to find a least squares solution to the problem of estimating the PA characteristic. However, this can not be done in the way presented in section 2.4 because of dependence between the columns in $\mathbf{X}$ in equation (2.50) [12]. Hence, the SVD[20] technique is used to solve the problem for the optimum coefficient vector $\hat{\mathbf{a}}$:

$$\hat{\mathbf{a}} = \mathbf{X}^+ \cdot \mathbf{z} \tag{2.57}$$

where $\mathbf{X}^+$ is the pseudoinverse of the matrix $\mathbf{X}$ and $\mathbf{z}$ is a vector containing the observed output baseband samples:

$$\mathbf{z} = [z_1 \quad z_2 \ \ldots \ z_N]^T \tag{2.58}$$

Notice that the solution contains no kind of weighting. The mathematical complexity of using weights when solving the problem using SVD is beyond the scope of this study. It is also important to notice that the ρ parameter of the recursive filter is not estimated in the SVD procedure. As described above, the expression from the thermal branch merely modifies the basis functions associated with the thermal branch compared to those of the top branch. Hence, ρ is chosen in advance. The optimal ρ is then found by a linear search, each time estimating the optimal $\hat{\mathbf{a}}$ with a different ρ. Now the modeled values will of course follow:

$$\hat{\mathbf{z}} = \mathbf{X}\hat{\mathbf{a}} \tag{2.59}$$

This completes the presentation of the empirical polynomial model.

2.6.2 The Empirical Spline Model

The empirical spline model is very much similar to the empirical polynomial model. The difference consists in a change of basis functions in the memoryless nonlinearities from polynomials to splines. Representing a memoryless nonlinearity by means of splines is presented thoroughly in section 2.4 and will not be repeated here. Figure 2.12 depicts the complete empirical spline model. As the figure shows, the only difference from the polynomial model is how the memoryless nonlinearities are modeled. All the signals in figure 2.10 will now be treated as baseband signals at once, and the tilde in $\widetilde{\mathbf{x}}$ etc. is ignored, typing $\mathbf{x}$, $\mathbf{y}$, $\mathbf{u}$ and so on. Hence, in the top branch we now have:

$$u_i = \sum_{q=-Q_1}^{Q_2} \sum_{j=K_0}^{K-1} h_q^{(rf)} a_j B_j(|x_{i-q}|) e^{j\angle x_{i-q}} = \sum_{q=-Q_1}^{Q_2} \sum_{j=K_0}^{K-1} a_{q,j} B_j(|x_{i-q}|) e^{j\angle x_{i-q}} \tag{2.60}$$

where $a_{q,j} = h_q^{(rf)} \cdot a_j$, with $h_q^{(rf)}$ being the baseband filter coefficients from the (discrete version of the) filter depicted in figure 2.10 and a_j being the spline coefficients. Of course, $a_{q,j}$ are treated as a (2-D) array, just as in the polynomial approach. The expressions for the second and third branches will not change with respect to the polynomial model. Thus, we have:

$$v_i = \sum_{l=0}^{l<i} |x_{i-l}|^2 \rho^l + \sum_{m=-M_1}^{M_2} h_m^{(bi)} |x_{i-m}| + 1. \tag{2.61}$$

[20] Singular Value Decomposition, see [12].

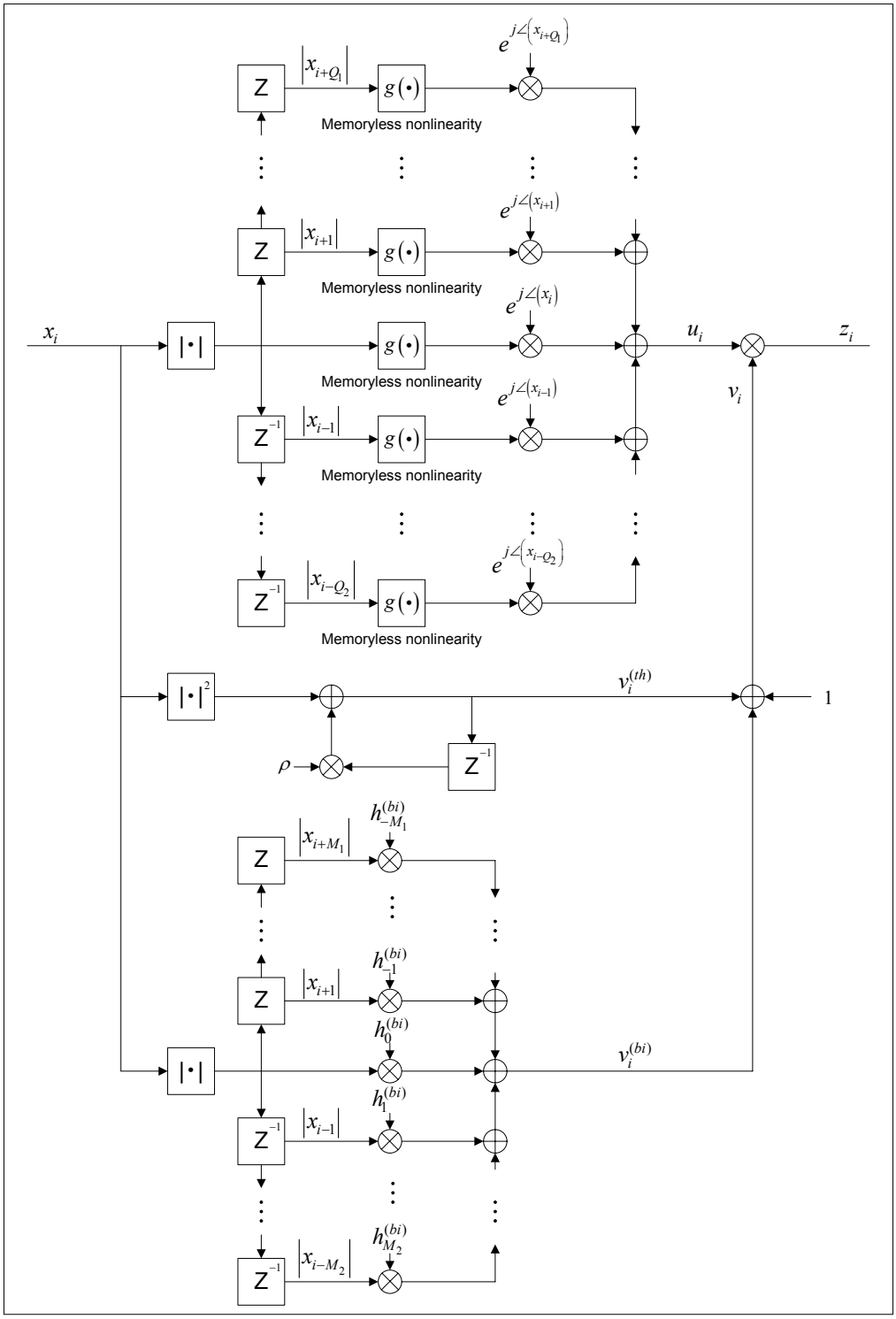


Figure 2.12: The empirical spline model.

Multiplying this by (2.60), the output from the empirical spline model is:

$$\begin{aligned} z_i = & \sum_{q=-Q_1}^{Q_2} \sum_{j=K_0}^{K-1} a_{q,j} B_j(|x_{i-q}|) e^{j\angle x_{i-q}} + \\ & \sum_{q=-Q_1}^{Q_2} \sum_{j=K_0}^{K-1} a_{q,j}^{(th)} B_j(|x_{i-q}|) e^{j\angle x_{i-q}} v_i^{(th)} + \\ & \sum_{q=-Q_1}^{Q_2} \sum_{j=K_0}^{K-1} \sum_{m=-M_1}^{M_2} a_{q,j,m}^{(bi)} B_j(|x_{i-q}|) e^{j\angle x_{i-q}} |x_{i-m}| \end{aligned} \tag{2.62}$$

Of course, $a_{q,j}^{(th)}$ and $a_{q,j,m}^{(bi)}$ are also treated as arrays over q and j, and q, j and m, respectively. Introducing the same uniform sparse delay-tap structure as in (2.47), the complete expression for the output from the empirical spline model is now:

$$\begin{aligned} z_i = & \sum_{q=-Q_1}^{Q_2} \sum_{j=K_0}^{K-1} a_{q,j} B_j(|x_{i-q'}|) e^{j\angle x_{i-q'}} + \\ & \sum_{q=-Q_1}^{Q_2} \sum_{j=K_0}^{K-1} a_{q,j}^{(th)} B_j(|x_{i-q'}|) e^{j\angle x_{i-q'}} v_i^{(th)} + \\ & \sum_{q=-Q_1}^{Q_2} \sum_{j=K_0}^{K-1} \sum_{m=-M_1}^{M_2} a_{q,j,m}^{(bi)} B_j(|x_{i-q'}|) e^{j\angle x_{i-q'}} |x_{i-m'}| \end{aligned} \tag{2.63}$$

where $q^{'}$ and $m^{'}$ follow (2.48) and (2.49). As in the previous subsection the output is now a linear function of the coefficients, which enables us to solve for the optimum coefficients using any least square algorithm. Switching to matrix notation now gives:

$$\mathbf{z} = \mathbf{X}\mathbf{a} \tag{2.64}$$

where $\mathbf{a}$ is a vector containing all the coefficients:

$$\mathbf{a} = \left[a_{-Q_1,K_0} \ \ldots \ a_{Q_2,K-1} \quad a_{-Q_1,K_0}^{(th)} \ \ldots \ a_{Q_2,K-1}^{(th)} \quad a_{-Q_1,K_0,-M_1}^{(bi)} \ \ldots \ a_{Q_2,K-1,M_2}^{(bi)} \right]^T \tag{2.65}$$

and

$$\mathbf{X} = \left[\mathbf{\Gamma}(x_1) \quad \mathbf{\Gamma}(x_2) \ \ldots \ \mathbf{\Gamma}(x_i) \ \ldots \ \mathbf{\Gamma}(x_N) \right]^T \tag{2.66}$$

where

$$\begin{aligned} \mathbf{\Gamma}(x_i) = [& \gamma_{-Q_1,K_0}^{(rf)}(x_i) \ \ldots \ \gamma_{Q_2,K-1}^{(rf)}(x_i) \quad \gamma_{-Q_1,K_0}^{(th)}(x_i) \ \ldots \\ & \gamma_{Q_2,K-1}^{(th)}(x_i) \quad \gamma_{-Q_1,K_0,-M_1}^{(bi)}(x_i) \ \ldots \ \gamma_{Q_2,K-1,M_2}^{(bi)}(x_i)]^T, \end{aligned} \tag{2.67}$$

$$\gamma_{q,j}^{(rf)}(x_i) = B_j(|x_{i-q'}|) e^{j\angle x_{i-q'}}, \tag{2.68}$$

$$\gamma_{q,j}^{(th)}(x_i) = B_j(|x_{i-q'}|) e^{j\angle x_{i-q'}} v_i^{(th)} \quad \Big(= B_j(|x_{i-q'}|) e^{j\angle x_{i-q'}} \sum_{l=0}^{l<i} |x_{i-l}|^2 \rho^l \Big), \tag{2.69}$$

$$\gamma_{q,j,m}^{(bi)}(x_i) = B_j(|x_{i-q'}|) e^{j\angle x_{i-q'}} |x_{i-m'}|. \tag{2.70}$$

Solving the problem using the SVD technique, the optimum coefficient vector $\hat{\mathbf{a}}$ is now:

$$\hat{\mathbf{a}} = \mathbf{X}^{+} \cdot \mathbf{z} \tag{2.71}$$

where $\mathbf{X}^{+}$ is the pseudoinverse of the matrix $\mathbf{X}$ and $\mathbf{z}$ is a vector containing the baseband output samples, just as in the empirical polynomial model. The same reasoning concerning the ρ parameter of the recursive filter mentioned above also applies here. Hence, the modeled values are now given by:

$$\hat{\mathbf{z}} = \mathbf{X}\hat{\mathbf{a}}. \tag{2.72}$$

This completes the empirical spline model. The procedure employed to find the optimum parameters K, Q_1, Q_2, ρ, M_1, M_2, $d^{(rf)}$ and $d^{(bi)}$ is the subject of section 3.4.

Chapter 3

Methodology

This chapter describes briefly how the models were implemented, and considers problems and limitations concerning this implementation. The data used in the modeling are presented along with actions that had to be taken to obtain well behaved results on count of the distribution of the data. The empirical models presented in section 2.6 present a problem of choosing the many parameters of the models. The procedure in which these were chosen is presented here. The chapter ends with a presentation of the different ways in which the results will be presented in chapter 4, along with a brief discussion on why these were chosen.

3.1 Implementation

All the code is written in MATLAB r2006a. The Digital Signaling Toolbox as well as the Optimization Toolbox were installed to solve the problems efficiently. All model implementations are based directly on the theory presented in chapter 2, although some modifications were done to make the code more efficient. However, these modifications do not conflict with any of the theory.

3.2 Problems and Limitations

When attempting to solve the kind of problems presented in this study, MATLAB is a very powerful tool. The only relevant problem faced is correctness with respect to the written code. It can sometimes be hard to validate if the written code is absolutely correct. Thus, proper testing has to be done. All the code used in this study has been tested extensively, showing no signs of flaws.

The only limitation faced was a few cases of memory overflow. As described in section 2.6, the empirical models have a lot of selectable parameters, giving them a large number of degrees of freedom. This sometimes led to memory overflow; too much data had to be processed at the same time. However, when discussing limitations, this happened because of RAM[1] limitations in the computer used in the simulations, not a limitation in MATLAB. This limitation did not however effect the results in ways that would have jeopardized the utility value of the study. This can be concluded as the overflow appeared at times when the performance tendencies were clear; a further increase in the number of coefficients would not lead to significant accuracy improvements.

[1] Random Access Memory.

3.3 The Data

Three different data sets have been used in the study. Their individual specifications will be presented after describing the equivalent model from which they were drawn. This model is depicted in figure 3.1. The vector **a** is a sequence of symbols to be transmitted. In continuous

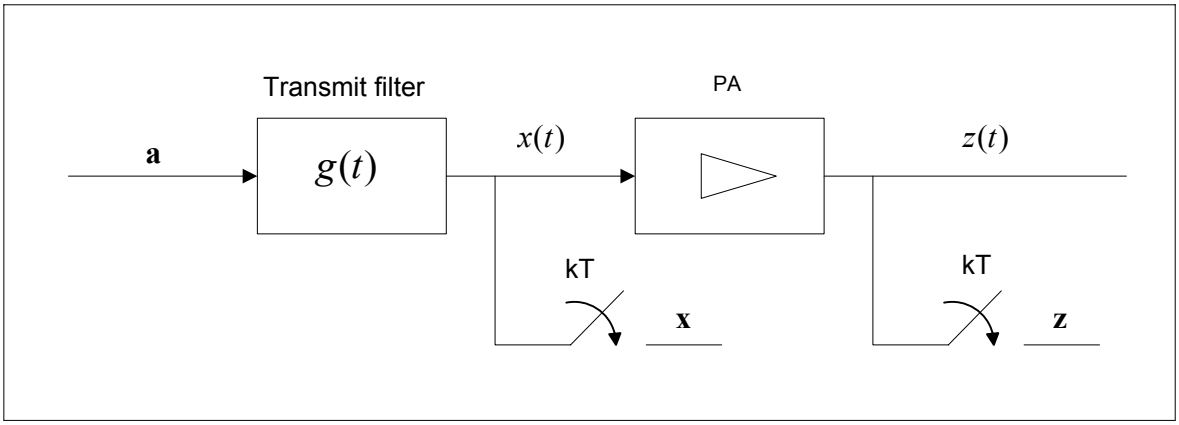


Figure 3.1: Model of data generation.

time this can be written as a sum of delta pulses:

$$a(t) = \sum_{i=0}^{N-1} a_i \delta(t - iT_s) \tag{3.1}$$

where a_i are the individual symbols and T_s is the time between two consecutive symbols; the symbol time. This pulse train is then sent through the transmit filter $g(t)$ with the resulting signal:

$$x(t) = \sum_{i=0}^{N-1} a_i g(t - iT_s) \tag{3.2}$$

The next block in the model is the PA. If the effect of the Pa is characterized as $f[\cdot]$, the output signal is:

$$z(t) = f[x(t)] \tag{3.3}$$

As the figure shows, both $x(t)$ and $z(t)$ are sampled to produce the two vectors **x** and **z**. Both these vectors contain the baseband samples, i.e. they are complex valued[2]. These are the samples used when modeling the PA. Note that the measurement noise mentioned in chapter 2 is not treated here, though all samples will of course contain some of this noise. The reason for presenting it explicitly in the model descriptions in chapter 2 was to show the precise relationship between the model output and the observations.

All three data sets come from 2 Watts PAs used in the Inmarsat Broadband Global Area Network (BGAN) terminals, each of them operated at different classes. The modulation employed in the BGAN system is 16-QAM. All sets have a signal bandwidth of about 190 kHz. The characteristics of the different data sets are as follows:

- BGAN1. Class A PA. Oversampling rate is 8. The set contains about 80 000 aligned input/output samples.

[2] $x(t)$ is in reality upconverted to its carrier frequency before passed through the PA, and then downconverted after the PA to produce $z(t)$. This is not shown in the figure.

- BGAN2. The biasing of the PA used is tunable for different class of operation. When obtaining this data set it was biased to class AB operation. The symbol rate of the transmit signal was 151.2 k samples/s. The sampling rate used when obtaining $\mathbf{x}$ and $\mathbf{z}$ was 25.5 MHz, leaving the oversampling factor to be a non-integer number of about 168.65. This set was decimated to an oversampling rate of 8 and contains about 45 000 aligned input/output samples.

- BGAN3. The same PA as in the BGAN2 case was used, now biased close to class B operation. The oversampling rate is thus 8 (after decimation) and the set contains about 45 000 aligned input/output samples.

All input and output samples are normalized[3].

3.3.1 Weighting the Data

One problem arises when looking at the distribution of the samples. Most of the sampels are taken close to the center of the interval $[0, 1]$, as depicted in figure 3.2. This is the distribution of

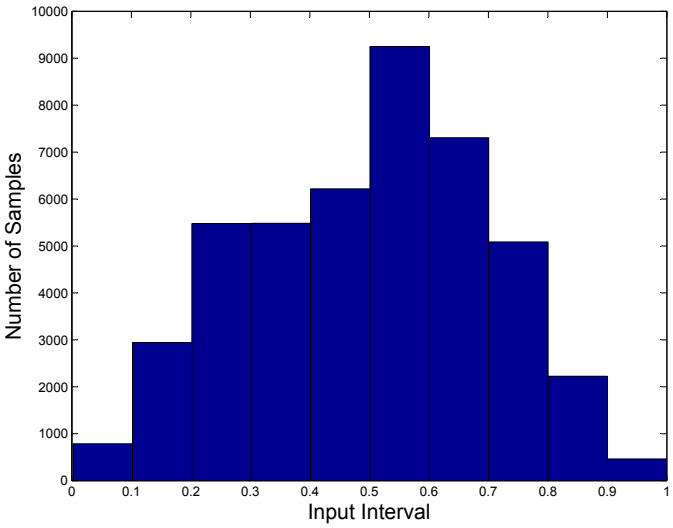


Figure 3.2: Data distribution.

the BGAN2 data set. The two other sets are similar in fashion. The large amount of data in the central part of the interval compared to the outer left and right parts will lead to a model highly dependent on this data. An example of such a scenario can be seen in figure 3.3. In this example a simulated PA characteristic is drawn in blue. 15 samples taken from this characteristic with added noise have been used to calculate a memoryless spline model for the PA, drawn in red. As the figure shows, with most of the samples situated in the central part of the interval, the calculated spline model is not accurate at all in the outer right part of the interval[4]. It is however accurate in the left part of the interval. This is because the PA characteristic is linear through approximately the starting three quarters of the interval, and so, few samples in the lower left part will not have influence on the final result. Hence, with the large amount of data in the

[3] $|x_i|, |z_i| \leq 1 \qquad \forall i$

[4] This is obviously an extreme example, only a single value (or multiple equal values) in the top right results in this approximation. It is nevertheless a nice example of the possible effects of this kind of distribution.

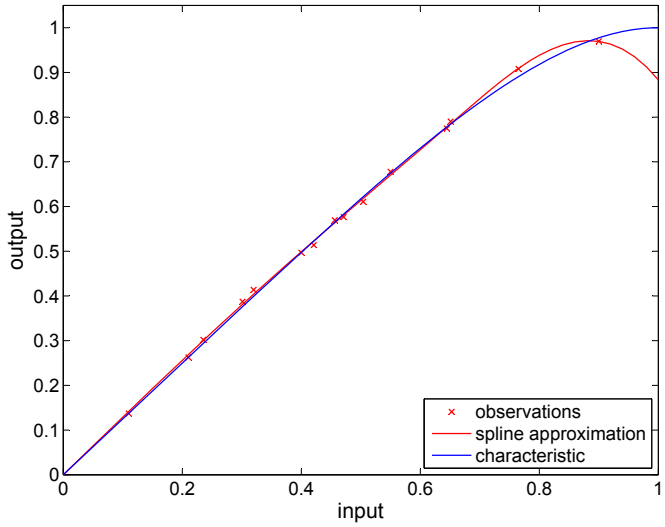


Figure 3.3: Example of data dependence.

central part of the interval in all three data sets, some sort of weighting is necessary. The main idea is to weight the errors in the outer right part of the interval more than the rest of the errors in equations (2.5) and (2.23). Remember that weighting of the data in the empirical models is not performed in this study due to the mathematical inconvenience. The type of weighting curve used is depicted in figure 3.4. As the figure shows the weights are approximately equal to 1 in

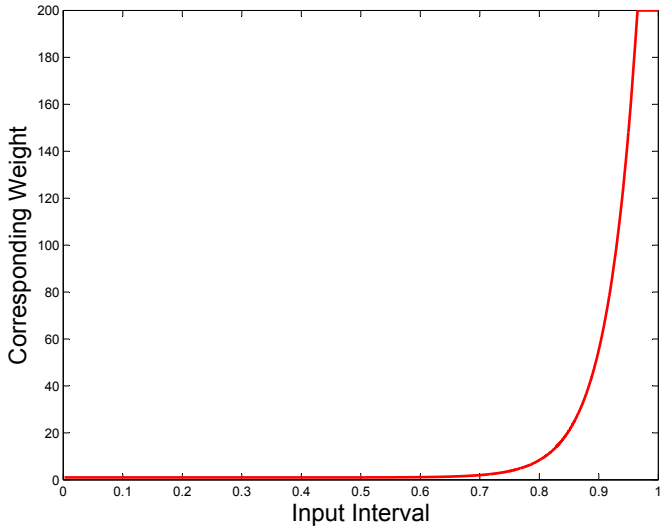


Figure 3.4: Weighting curve.

the first three quarters of the interval. In the last quarter the weights are larger[5]. This ensures accuracy in the entire interval, not just in the first three quarters.

[5]No weights are allowed to be larger than 200.

3.3.2 Windowing the Data

For systems with memory, modeling approaches as presented in this study have four different ways of windowing the data; the covariance method, the autocorrelation method, the prewindowing method and the postwindowing method [17]. Consider a very simple example. One wishes to estimate the coefficients of a simple three-tap FIR filter based on a least square solution. The input to the filter is $\mathbf{x}$ and the output is $\mathbf{y}$, both vectors known:

$$\mathbf{x} = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5] \tag{3.4}$$

$$\mathbf{y} = [y_1 \quad y_2 \quad y_3 \quad y_4 \quad y_5] \tag{3.5}$$

The modeled output is $\hat{y}_i = h_{-1}x_{i+1} + h_0 x_i + h_1 x_{i-1}$. The problem is now that the lack of a x_0 and x_6 prevents a complete calculation of $\hat{y}_1$ and $\hat{y}_5$, respectively. The different methods solve this problem in the following ways:

- The covariance method. This method insists on calculating only complete expressions when producing $\hat{\mathbf{y}}$. This means that no assumptions about the data outside the interval $[1, 5]$ are made. Consequently, only $\hat{y_2}$, $\hat{y_3}$ and $\hat{y_4}$ can be calculated when employing this method. $\hat{y}_1$ cannot be calculated due to the lack of x_0 and $\hat{y}_5$ cannot be calculated due to the lack of x_6. The vectors used in the estimation when employing the covariance method are hence:

$$\mathbf{y} = [y_2 \quad y_3 \quad y_4]$$

$$[x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5] * [h_{-1} \quad h_0 \quad h_1] \overset{Covar.}{\rightarrow} [\hat{y}_2 \quad \hat{y}_3 \quad \hat{y}_4]$$

- The autocorrelation method. This method uses zero-padding in the input vector $\mathbf{x}$ to produce a modeled $\hat{\mathbf{y}}$ with the same length as $\mathbf{y}$. Hence, the assumption that all samples prior to x_1 and after x_5 are zero is made. The model coefficients are then estimated based on all the values in $\mathbf{y}$. The vectors used in the estimation when employing the autocorrelation method are hence:

$$\mathbf{y} = [y_1 \quad y_2 \quad y_3 \quad y_4 \quad y_5]$$

$$[0 \quad x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad 0] * [h_{-1} \quad h_0 \quad h_1] \overset{Autocorr.}{\rightarrow} [\hat{y}_1 \quad \hat{y}_2 \quad \hat{y}_3 \quad \hat{y}_4 \quad \hat{y}_5]$$

- The prewindowing method. This method assumes that the samples prior to x_1 are zero, but makes no assumptions about the samples after x_5. The vectors used in the estimation when employing the prewindowing method are hence:

$$\mathbf{y} = [y_1 \quad y_2 \quad y_3 \quad y_4]$$

$$[0 \quad x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5] * [h_{-1} \quad h_0 \quad h_1] \overset{Prewin.}{\rightarrow} [\hat{y}_1 \quad \hat{y}_2 \quad \hat{y}_3 \quad \hat{y}_4]$$

- The postwindowing method. This method is the opposite of the prewindowing method, assuming zero samples after x_5, but making no assumptions about the samples prior to x_1. The vectors used in the estimation when employing the postwindowing method are hence:

$$\mathbf{y} = [y_2 \quad y_3 \quad y_4 \quad y_5]$$

$$[x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad 0] * [h_{-1} \quad h_0 \quad h_1] \overset{Postwin.}{\rightarrow} [\hat{y}_2 \quad \hat{y}_3 \quad \hat{y}_4 \quad \hat{y}_5]$$

The zero-padding used in the three latter methods might influence the results in the estimation procedure. Hence, the covariance method is used in this study to avoid any possible effects of zero-padding. In the example above only 5 input/output samples are available in the estimation of 3 coefficients. Using the covariance method would only leave $\hat{y}_2$, $\hat{y}_3$ and $\hat{y}_4$ vs y_2, y_3 and y_4 for the estimation procedure. This is of course not enough in order to estimate the 3 coefficients with high accuracy. However, this kind of scenario will not occur in the modeling procedures presented in this study, as the number of input/output samples in the data sets exceeds the number of model coefficients by far. Thus, choosing the covariance method causes no problem of this kind. To give a specific example concerning this study, consider the Hammerstein model presented in section 2.5. The vector $\mathbf{z}$, containing the observed output data, is now truncated as follows:

$$\mathbf{z} = [z_{L+1} \quad z_{L+2} \ \dots \ \quad z_{N-L}] \tag{3.6}$$

The Hammerstein model will of course produce $N - 2L$ values (at each iteration), modeling the components of (3.6).

3.4 Selecting the Parameters

The models presented in chapter 2 have various numbers of parameters that have to be decided. The memoryless and the Hammerstein model both have a very limited number of parameters[6], and a specific procedure to select these is thus of no concern. However, the two empirical models presented in section 2.6 have a large number of parameters (also referred to as degrees of freedom). Hence, a way of choosing these parameters in an optimal fashion is needed[7]. This is the topic of this section. In the beginning of section 2.6 it was made clear that the number of coefficients in the model is of no concern in this study. When selecting the parameters, this is not completely true. As described in 3.2, memory limitations have an impact on the number of parameters. Hence, the number of coefficients and the achieved model accuracy has to be balanced with respect to the memory limitations. The following procedure presents how the parameters of both the empirical spline and the empirical polynomial model were chosen. For each step, the best parameter set[8] from the previous step is used. The accuracy criterion used in all the steps is the mean square error (MSE), discussed in section 3.5.

1. Choosing K, the number of intervals in the spline model and the maximum order of the polynomial in the polynomial model. In the spline model K is simply chosen based on the results obtained when testing the memoryless model. For the case of the polynomial model K is chosen based on the results presented in [4].

2. Choosing the memory parameters Q_1, Q_2 and $d^{(rf)}$. At this point, testing whether or not to employ the origin constraint[9] in the spline model is also chosen. The accuracy of the models with several combinations of the parameters is calculated. Q_2 is expected to be larger than Q_1, as it is natural that more memory effects come from previous samples than from future samples. This fact is taken into account when choosing Q_1 and Q_2.

[6] The number of intervals (K) for the memoryless model and, in addition to this, the memory parameter (L) for the Hammerstein model.

[7] With this many parameters, an exhaustive search over all combinations to find the optimum solution is not feasible.

[8] The best set now means the set from which an expansion only results in an insignificant accuracy improvement.

[9] $K_0 = -3$ or $K_0 = -1$ (plus deleting B_{-2} and B_{-3} and modifying B_{-1}).

3. Choosing the multiplication parameter ρ in the recursive filter. This is a pretty simple step. Adding the thermal effect to the best models from the previous step, and finding the optimal ρ.

4. Choosing the bias parameters M_1, M_2 and $d^{(bi)}$. Selecting these is done in the same way as how the memory parameters were chosen in step 2. However, M_2 (and to a lesser degree M_1) is typically a greater number than Q_2 (Q_1).

Figure 3.5 depicts a simple flowchart of the parameters selection.

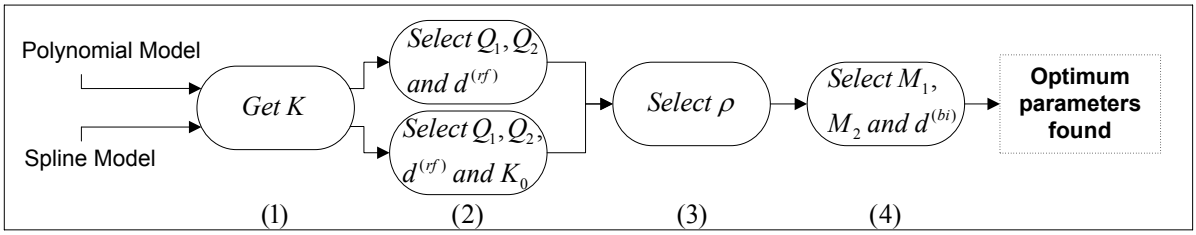


Figure 3.5: Simple flowchart of parameter selection.

3.5 Results Presentation

In order to measure the accuracy of the PA models, the results obtained have to be evaluated. This is done by comparing the estimated signal at the output of the model with the observed PA output, both in time- and frequency-domain. Figure 3.6 depicts how the different evaluations are done.

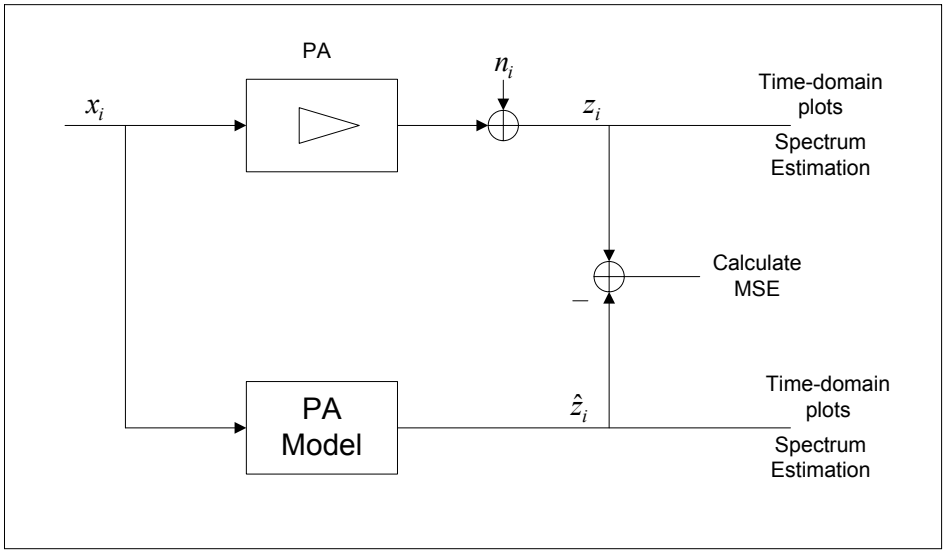


Figure 3.6: Block diagram of PA model evaluation.

In the time-domain the Mean Squared Error in decibel is employed to evaluate the modeling accuracy:

$$MSE^{(dB)} = 10log_{10}\left(\frac{1}{N}\sum_{i=1}^{N}|z_i - \hat{z}_i|^2\right) \tag{3.7}$$

where z_i are the observed output baseband samples, $\hat{z}_i$ are the modeled samples and N is the number of samples. This is a well known accuracy measurement which makes it easy to compare the results obtained in this study with other studies in the same field. Also, different kinds of plots are presented in the time-domain. For the memoryless case, the modeled PA characteristic is drawn along with the observed values, both the AM/AM and the AM/PM curves. This gives a nice visual idea of how accurate the estimated model is. When memory effects are taken into account, these types of plots are no longer applicable. We now plot the modeled values versus the observed values, one plot for the real parts and another for the imaginary parts. If the model estimated was perfect ($\hat{z}_i = z_i \quad \forall i$), these curves would be straight lines, as depicted in figure 3.7 for the real case. This will of course not be the case, but the plots are nevertheless interesting in

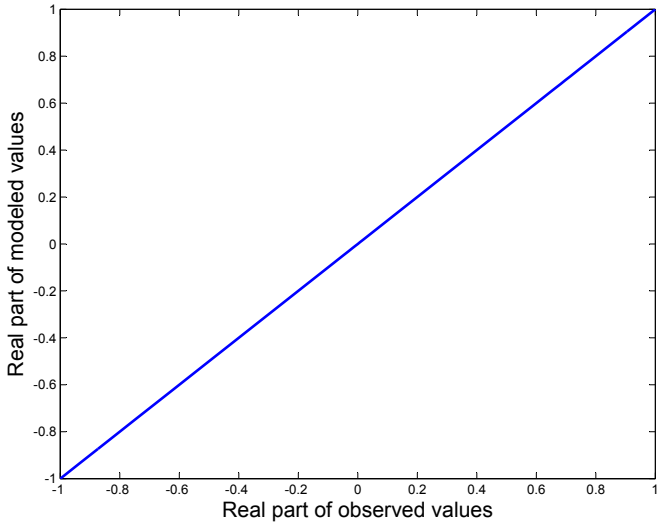


Figure 3.7: Perfect PA modeling.

the context of evaluating the accuracy. The closer the plots are to a straight line, the better the accuracy of the model. These plots are also a way to evaluate whether the modeling is consistent or not. If both the real and the imaginary plots shows nice (linear) behaviour, the modeling of the PA characteristic works, both in amplitude and phase. If one (or both) of them are flawed, the model has failed to estimate the characteristic.

In the frequency-domain, estimation of the power spectral density (PSD) of the signal is a well-known accuracy evaluation criterion. The PSD is estimated using the *pwelch* function in MATLAB. This function employs the Welch method of averaging modified periodograms in the spectrum estimation, thoroughly described in [9]. The PSD of the observed output signal is plotted along with the PSDs of various modeled outputs. This is a nice way of displaying the accuracy of each model with respect to both the actual PA output and the other models. It is important to notice that the spectrum estimation procedure used is not very accurate in the outer part of the spectrum, so large differences in these areas should not lead to a conclusion of bad model accuracy.

Chapter 4

Results

This chapter presents the results of the study. Each model presented in chapter 2 has its own section, with the corresponding results and analysis. The figures are labeled BGAN1, BGAN2 and BGAN3, in accordance with the data sets described in section 3.3.

4.1 The Memoryless Model

The most important results when dealing with the memoryless model are the AM/AM and the AM/PM curves, described in section 3.5. The reason why only the amplitude is considered in the input signal is (as described earlier) that the PA characteristic is independent of the input phase when assuming no memory. All three data sets performed well in the modeling of the characteristics. Shown in figure 4.1 are the AM/AM and AM/PM curves for all three data sets[1] drawn together with the actual observations.

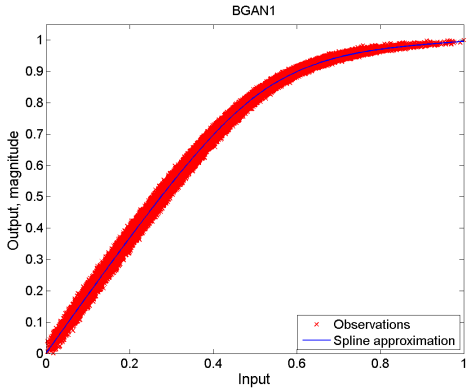
The three PAs obviously have different characteristics, as expected. The spline approximation models the characteristics nicely; it is clear that both the AM/AM and the AM/PM curves fit nicely with the actual observed values. One might say that the AM/PM curves do not perform well when considering the pretty large errors that occur in the lower regions of the input. However, this is actually not a problem because at these small input amplitudes, a phase error will not induce serious damage on the approximated output signal. The reason for this is depicted in figure 4.2, along with the impact a phase error of this sort would have for larger input amplitude. The figure shows the impact on two different signals for a phase shift of about $\frac{\pi}{6}$ radians, indicated by the arrow. The left part of the figure shows the effect on a signal with small amplitude and the right part for a signal with large amplitude. The red line is the error vector. The error is relatively small when the input amplitude is small. For the considerably larger input amplitude, the error vector is quite large. As the phase errors in the memoryless models are large only for small input amplitudes, this is thus no problem.

As described in section 3.5 it is also instructive to inspect the mean squared error between the actual observations and the modeled values. Table 1 presents this data. As the table shows

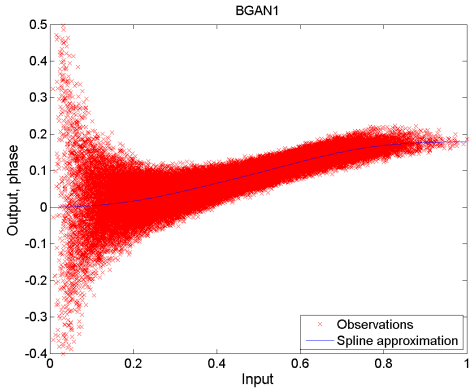
PA	**MSE [dB]**
BGAN1	-34.9
BGAN2	-29.4
BGAN3	-26.3

Table 4.1: MSE memoryless models.

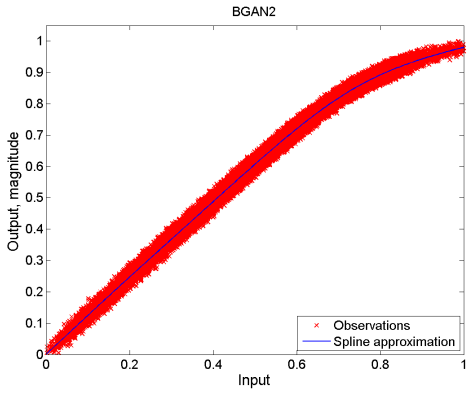
[1] Phase in radians.



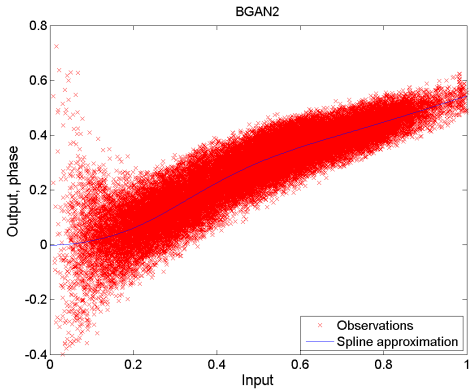
(a) AM/AM BGAN1.



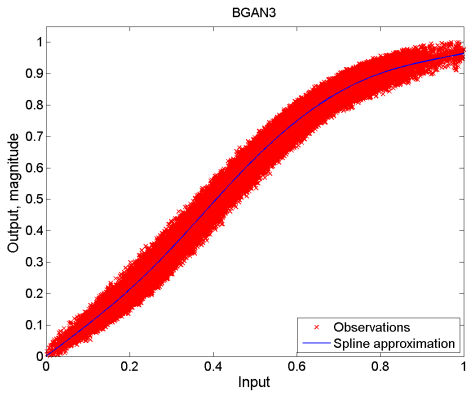
(b) AM/PM BGAN1.



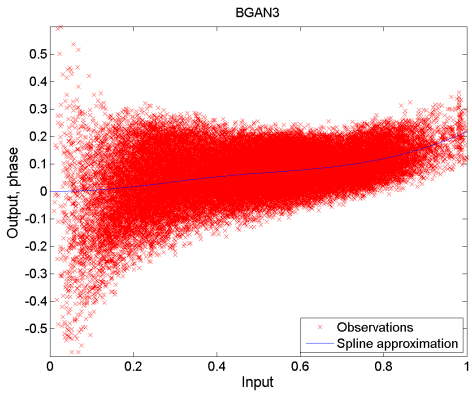
(c) AM/AM BGAN2.



(d) AM/PM BGAN2.



(e) AM/AM BGAN3.



(f) AM/PM BGAN3.

Figure 4.1: Memoryless model characteristics.

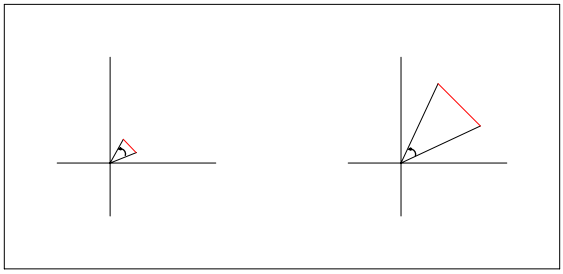


Figure 4.2: Phase error impact.

the MSE is below -26 dB for all three models. These results show that the memoryless spline model works quite well indeed. However, as will be clear later, these results do not provide the same accuracy as do more complex models. The results presented above were all obtained using 4 intervals in the spline approximation. When working with splines in this kind of context, dividing into more intervals, we would expect better performance [3, 11]. More intervals lead to a more accurate description, but also increased complexity and slower convergence time, as it results in the computation of more coefficients. Thus we have a performance/complexity tradeoff situation. In section 2.6 it was made clear that the number of coefficients is only of limited concern when dealing with the models of this study. These empirical models are far more complex than the memoryless model, and discussing a performance/complexity tradeoff situation at this point might seem a little superfluous; the number of intervals will in no way cause memory limitation problems (see 3.2 and 3.4). However, it is important to have mentioned that such a tradeoff exist, as it may be important in other situations. Hence, the models were tested with a various number of intervals. Figure 4.3 depicts the MSE for all three data sets as a function of the number of intervals in the spline approximation. The figure shows the existence of the

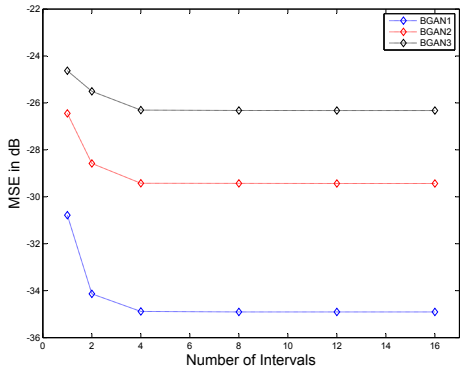
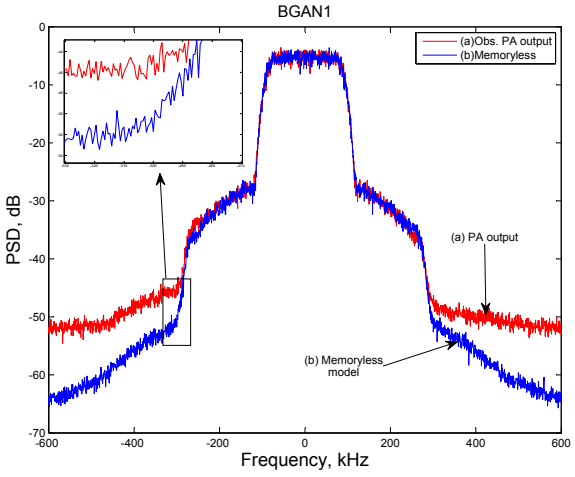


Figure 4.3: MSE as a function of the number of intervals.

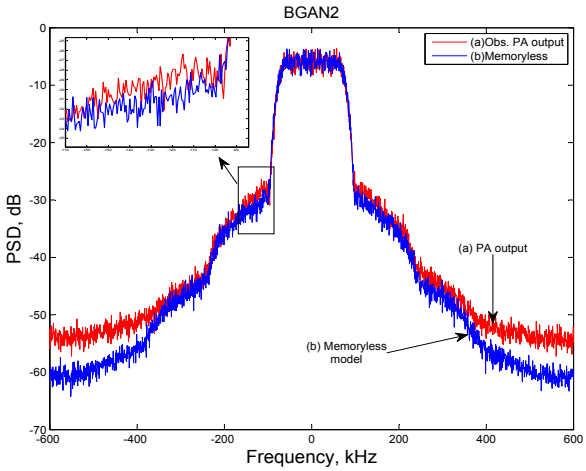
relationship described above; as the number of intervals increases the MSE decreases. However, it is clear from the figure that employing more than 4 intervals in any of the models will have negligible effects on the accuracy. Thus, the results presented in this section all use 4 intervals in the spline approximation.

The difference in MSE between the three PA models can be partially explained by inspecting the AM/AM and AM/PM curves in figure 4.1. The variance in both the amplitude and the phase response of the observed values in data sets BGAN2 and BGAN3 are higher than what is the case in data set BGAN1. This will of course result in a higher MSE.

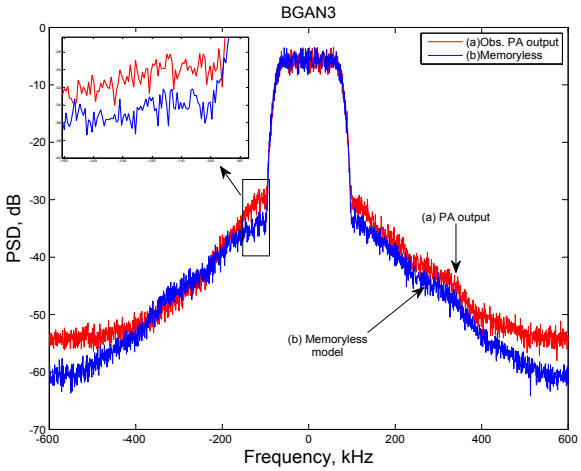
The estimated PSDs of the three memoryless model outputs drawn along with the PSDs of the respective PA outputs are depicted in figure 4.4. The main lobe is well estimated in all three



(a) BGAN1 PSD.



(b) BGAN2 PSD.



(c) BGAN3 PSD.

Figure 4.4: Memoryless model PSDs.

cases. The accuracy differences between the cases are obvious, it is clear that the model PSD in the BGAN1 case follows the PA output PSD much better than what is observed in the two other cases. The modeling problems for the BGAN1 case starts in the second sidelobe, as depicted in the figure. The level of this sidelobe relative to the mainlobe is very low (approximately 40 dB below), and the difference between the PSD in this area does not imply an inaccurate model. For the BGAN2 and BGAN3 cases the problems start in the first sidelobe, resulting in less accurate models. At this point it is important to remember that the models were in no way designed to minimize the estimated PSD between the observed output and the modeled output[2].

[2]Minimization criterion used was MSE.

4.2 The Hammerstein Model

This section presents the results obtained when employing the Hammerstein model. All results presented were obtained using 4 intervals in the memoryless spline part of the model, based on the results from the previous section.

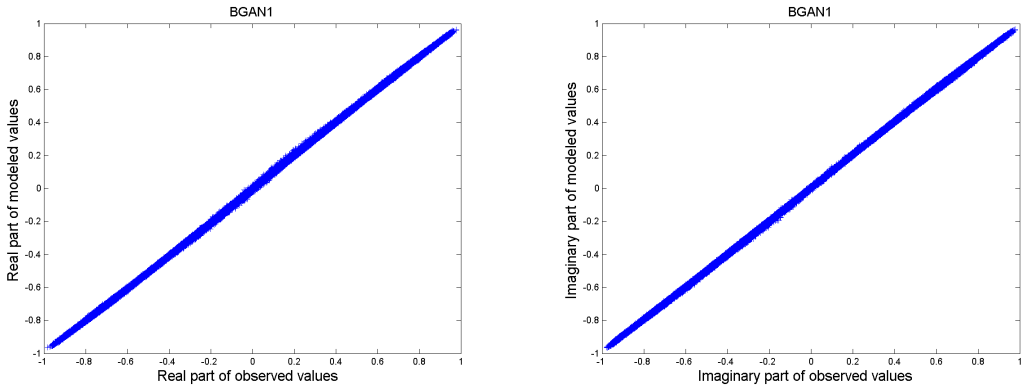
BGAN1

The MSE results obtained when modeling the BGAN1 amplifier are presented in table 4.2. The

Filter Length	MSE [dB]
0 (no memory)	-34.883
3	-40.275
5	-40.140
7	-39.868
9	-39.413

Table 4.2: MSE Hammerstein model, BGAN1.

result obtained when using no filter ($L = 0$) is presented for comparison purposes. As the table shows, when inserting a filter of length 3, an accuracy improvement close to 5.4 dB is observed. However, when expanding the filter further, the MSE increases. These are somehow contradicting results; with such a large improvement when moving from no filter to a simple three-tap filter, we would expect a further accuracy improvement when expanding the filter. One of the main reasons why this does not happen is the weighting procedure described in section 3.3. The weighting makes sure that the model is as accurate as possible in the entire input interval. Thus, the MSE is not minimized when weighting the errors. Calculation of the MSEs for the cases in table 4.2 when not employing weighting of the errors were also done in the study, showing an accuracy improvement of about 0.2 dB when going from filter length 3 to filter length 9. This supports what was stated above concerning the effect of the weighting on the results. Figure 4.5 depicts the real and imaginary parts of the observed and the modeled values, plotted as described in section 3.5, for the best case obtained (filter length 3). It is clear from the figure that both the



(a) Real parts.

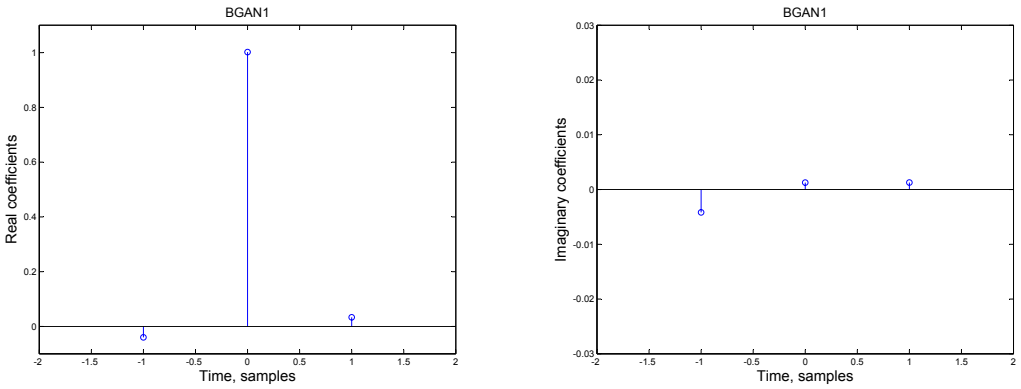
(b) Imaginary parts.

Figure 4.5: Real and imaginary plots, BGAN1.

amplitude and phase response of the PA is nicely modeled, as both the real and the imaginary

curves are well behaved. Both curves are indeed not far away from a straight line as the one depicted in figure 3.7. This demonstrates once again a very accurate model.

It is also of great interest to inspect the filter coefficients calculated. Figure 4.6 depicts the real and the imaginary parts of the filter coefficients. As the figure shows the filter is almost



(a) Real part.

(b) Imaginary parts.

Figure 4.6: Real and imaginary filter coefficients, BGAN1.

completely real, the imaginary parts are approximately equal to zero. Figure 4.7 depicts the magnitude of the filter coefficients. This shows that the two taps modeling the memory is only in

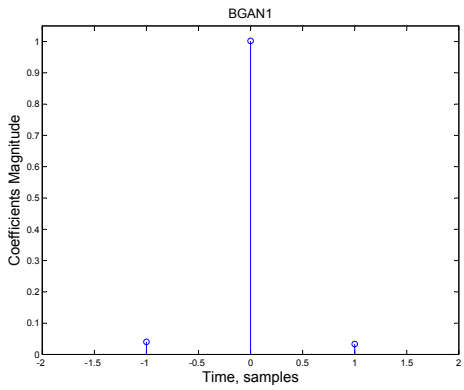
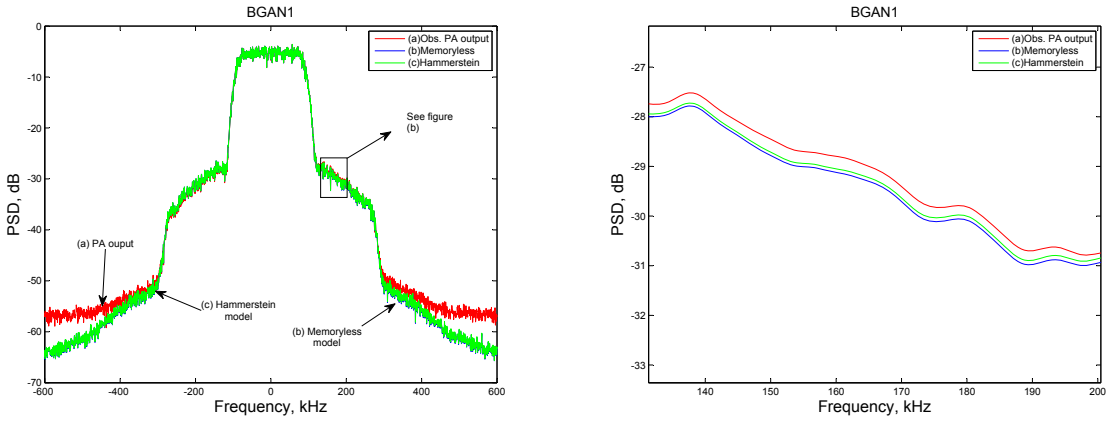


Figure 4.7: Magnitude of filter coefficients, BGAN1.

the order of 3-4 % of the centertap, which is approximately equal to 1. Hence, quite small valued memory taps can improve the accuracy compared to the memoryless approach considerably.

Depicted in figure 4.8 are the estimated PSDs for the BGAN1 observed output, memoryless model and Hammerstein model. It is not easy to see the difference between the Hammerstein and the memoryless model in figure 4.8a. Thus, the segment marked by the rectangle is recalculated using a smaller window size in the estimation of the PSD. This smoothes the curve and makes it easier to observe the differences. Figure 4.5b shows this segment. Now the superiority of the Hammerstein model compared to the memoryless model is clear. The segment is localized in the area where the Hammerstein PSD and the memoryless PSD begins to deviate. Hence, the accuracy improvement is first noticed in the first sidelobe of the PSD.



(a) BGAN1 PSD.

(b) Segment of PSD.

Figure 4.8: Figure (a) depicts the PSD calculated. Figure (b) depicts the segment marked in (a) using a smaller window size. This smoothes the curve and makes it easier to observe the differences.

BGAN2

The MSE results obtained when modeling the BGAN2 amplifier are presented in table 4.3. As

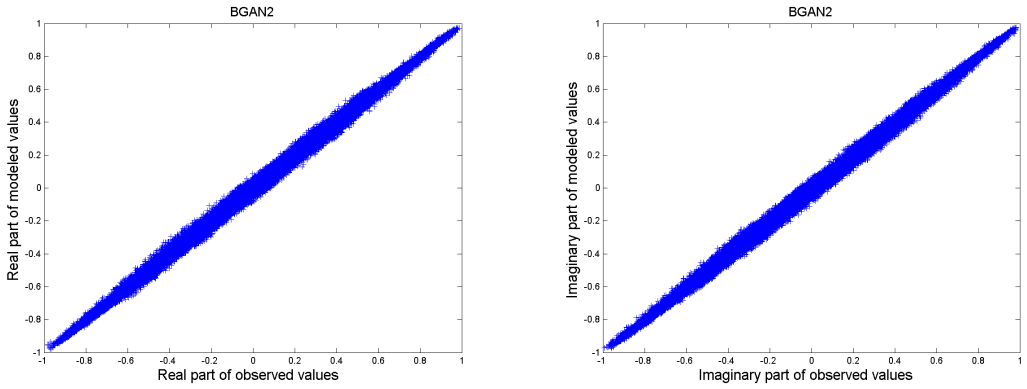
Filter Length	MSE [dB]
0 (no memory)	-29.424
3	-29.429
5	-29.401
7	-29.407
9	-29.401

Table 4.3: MSE Hammerstein model, BGAN2.

the table shows, no accuracy improvements compared to the memoryless model were obtained[3]. These are quite disappointing results indeed when considering the significant improvement found when modeling the BGAN1 PA. However, as will be clear in later sections, memory effects can originate in different ways in different PAs. Estimating models without employing weighting were also done for the BGAN2 case, resulting in an accuracy improvement of about 0.15 dB when using a filter of length 7. This is of course nothing compared to the improvements in the BGAN1 case, but it is nevertheless an improvement. Presented in figure 4.9 are the real and imaginary plots obtained with filter length 3. These plots are presented only to show that the Hammerstein model obtained well behaved results both for the amplitude and phase response of the BGAN2 PA, even though no accuracy improvements were found. As the figure shows, both the real and the imaginary plots possess the basic nature of a straight line. However, it is clear that the variance is much greater than the corresponding plots for the BGAN1 PA shown in figure 4.5, explaining the large accuracy difference. The corresponding filter is approximately a delta pulse. Thus, it is not necessary to grant it with a figure.

The lack of accuracy improvement means that the estimated PSD of the Hammerstein model will equal the estimated PSD of the memoryless model. Figure 4.4b presents this information

[3] A 0.005 dB improvement is not considered as an actual improvement, rather as an effect of e.g. the windowing method employed, see 3.3.



(a) Real parts. (b) Imaginary parts.

Figure 4.9: Real and imaginary plots, BGAN2.

with the Hammerstein PSD coinciding with the memoryless PSD depicted. Thus, it is not presented here.

BGAN3

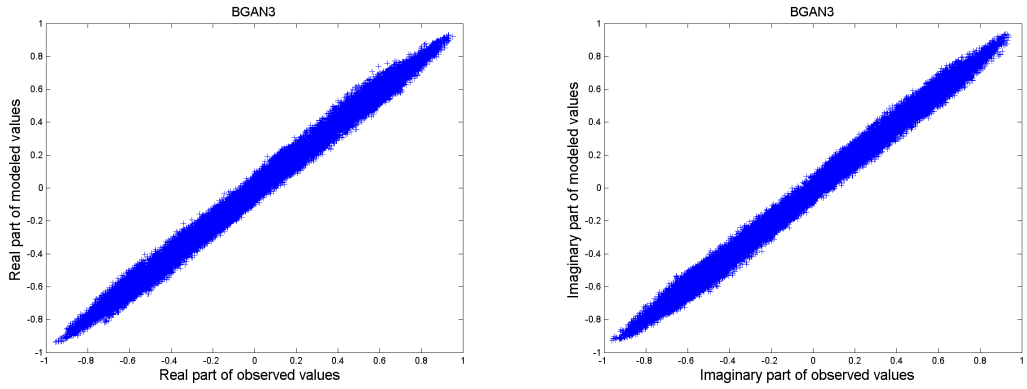
The MSE results obtained when modeling the BGAN3 amplifier are presented in table 4.4. The

Filter Length	MSE [dB]
0 (no memory)	-26.306
3	-26.307
5	-26.306
7	-26.309
9	-26.309

Table 4.4: MSE Hammerstein model, BGAN3.

table shows that the results obtained here follow the same trend as the results concerning the BGAN2 PA; moving from the memoryless model to the Hammerstein model does not result in any accuracy improvement. This was however no big surprise as the two data sets, BGAN2 and BGAN3, are obtained from the same PA, only differing in the class of operation. Thus, when experiencing no accuracy improvements when investigating BGAN2, one would expect a similar trend when investigating BGAN3. The limiting effects of the weighting used were of course also present now, and so, Hammerstein models were estimated using no weighting. These estimations showed an accuracy improvement of about 0.1 dB when employing a filter of length 7. This improvement is not significant at all when comparing it to the results obtained with the BGAN1 data set, but, as for the BGAN2 case, it is nevertheless an improvement. Depicted in figure 4.10 are the real and imaginary plots obtained with filter length 3. They are, as for the BGAN2 case, presented to show that the results are well behaved both for the amplitude and the phase response. Also now both plots possess the basic nature of a straight line. The variance is however even larger now compared to the BGAN2 plots, explaining the higher MSE. Of course, the filter calculated is also now approximately equal to a delta pulse and is not granted with a figure.

The estimated PSD are not presented here, with the same reasoning as in the BGAN2 case above. Figure 4.4c displays this information with the Hammerstein PSD coinciding with the memoryless PSD.



(a) Real parts.

(b) Imaginary parts.

Figure 4.10: Real and imaginary plots, BGAN3.

General Analysis

The iterative coefficient estimation procedure employed presented some problems. As the results show, no accuracy improvements compared to the memoryless model were obtained in the BGAN2 and BGAN3 cases. However, the reasons behind these difficulties are hard to explain. Whether or not the error functions are strictly convex is unknown, and it is thus not possible to know if the estimation procedures converged to local or global minimums. An attempt to investigate this was made by employing other initial vectors than the one presented in section 2.5. These new vectors varied in the range from an all-zero vector to slightly modified versions of the presented initial vector. All attempts led to the same solution with no accuracy improvement. Thus, the convexity of the error functions in these two cases is assumed strict with no reasonable doubt. This assumption was further strengthened by the fact that the empirical spline model obtained the very same results when structured as a Hammerstein model[4]. This means that the Hammerstein model is simply not fit to improve the modeling accuracy for the BGAN2 and BGAN3 PAs. Hence, the problem lay not in the estimation procedure, but in the applicability of the Hammerstein model.

4.3 The Empirical Model

This section presents the results obtained when employing the two empirical models outlined in section 2.6. The results from the three PAs are presented separately, and the results from the empirical polynomial and the empirical spline model for each PA are presented together. This renders comparisons between the models as a possibility. The parameters employed when obtaining the results follows the procedure outlined in section 3.5. To be able to see the influence of the different memory effects, the tables present the optimum results for each step in the flowchart depicted in figure 3.5. As described the parameter K, setting the number of intervals in the spline model and the maximum order of the polynomial in the polynomial model, are chosen based on previous results in this study and in [4], respectively. Thus $K = 4$ in the spline model and $K = 7$ in the polynomial model. All results presented for the empirical spline model

[4] $Q_1 = Q_2 = L$ and $d^{(rf)} = 1$ in the parameter selection. As described in section 2.6 the model is not a proper Hammerstein model, but can nevertheless be used for comparison purposes. The results obtained were equal to the results from the Hammerstein model with no weighting.

employs the origin constraint, as omitting this constraint (and thus increasing the complexity) provided no accuracy improvements.

BGAN1

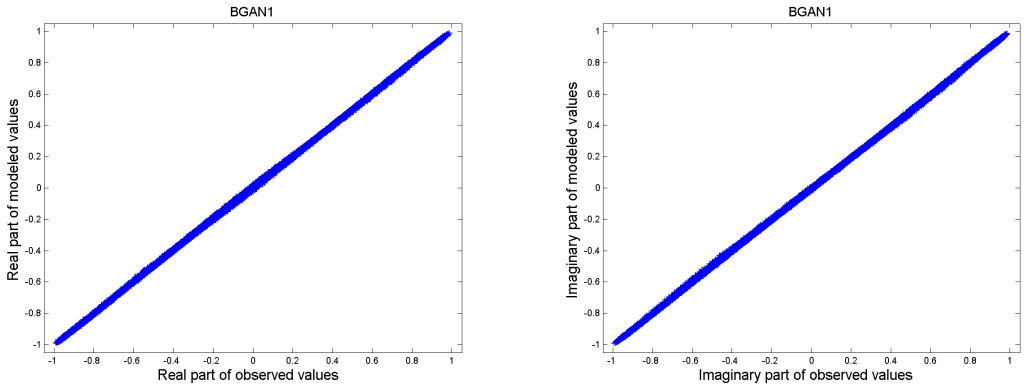
The parameter optimization procedure presented in section 3.4 resulted in equal parameters for both models. Hence, the results can be presented together in a single table. These are presented in table 4.5. The results from the two models obviously follow the same pattern. Only in the

PA model parameters	**MSE [dB]**	
	Polynomial	**Spline**
Memoryless	-34.812	-34.890
$Q_1 = 1$, $Q_2 = 2$, $d^{(rf)} = 1$	-40.317	-40.598
$Q_1 = 1$, $Q_2 = 2$, $d^{(rf)} = 1$ $\rho = 0.5$,	-40.818	-40.869
$Q_1 = 1$, $Q_2 = 2$, $d^{(rf)} = 1$ $\rho = 0.5$, $M_1 = 1$, $M_2 = 4$, $d^{(bi)} = 1$	-41.074	-41.091

Table 4.5: MSE empirical models, BGAN1.

second row, including only the top branch in figure 2.10, an accuracy difference of approximately 0.3 dB can be said to favour the spline model in some sense. The reason why the results are so similar will be discussed in the general analysis below. We now analyze the performance of the general empirical model, as the two variations of it show such similar behaviour. It is clear that in this case the top branch, initially modeled as a Hammerstein model, contributes significantly to the accuracy improvement. This was also expected based on the results in the previous section. However, only a small improvement was seen when introducing the two lower branches in the model. The thermal effects provided a 0.3-0.5 dB improvement and the biasing effects another quarter of a dB. Thus, the resulting accuracy improvement is from approximately 0.5 to 0.75 dB. In other words not a vast improvement, but nevertheless an improvement. Other approaches have to be considered if one wishes significant improvements beyond those provided by the top branch.

Figure 4.11 depicts the real and imaginary plots for the optimal spline model presented in table 4.5. Since the results from the two models are so similar it is of no use to plot the



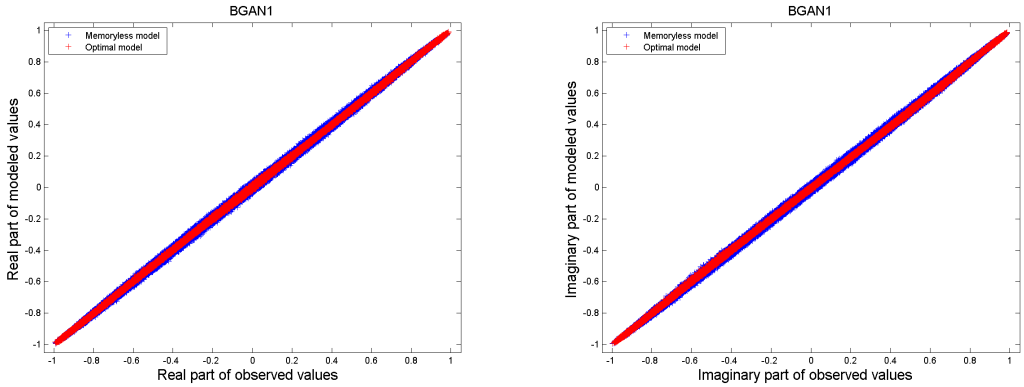
(a) Real parts.

(b) Imaginary parts.

Figure 4.11: Real and imaginary plots for the optimal empirical spline model, BGAN1.

corresponding polynomial model curves. They have however been inspected, showing the same

behaviour. Both curves possess the nature of a straight line, showing that the amplitude as well as the phase response of the PA are modeled properly. It is interesting to plot the same curves for the memoryless model along with the curves for the optimal model. This gives a nice visual impression of the accuracy improvement. This is depicted in figure 4.12, also here only presenting the spline model results. The accuracy improvement is clearly seen in the figure; the

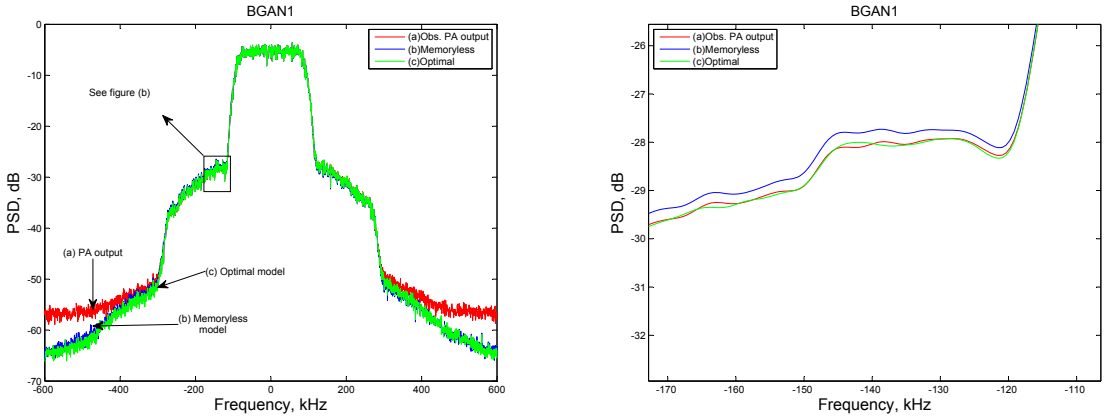


(a) Real parts. (b) Imaginary parts.

Figure 4.12: Memoryless model vs. optimal model, BGAN1.

variance of the optimal curves compared to a straight line is less than what the case is for the memoryless model.

Now we inspect the estimated PSDs of the models and compare them with the estimated PSD of the observed output. Figure 4.13 depicts these PSDs. Again only the results from the



(a) BGAN1 PSD. (b) Segment of PSD.

Figure 4.13: Figure a depicts the PSD calculated. Figure b depicts the segment marked in (a) using a smaller window size. This smooths the curve and makes it easier to observe the differences.

spline modeling are depicted, as the results from the polynomial model are so similar. It is hard to see the superiority of the optimal model compared to the memoryless model in figure 4.13a. Thus, the same technique used in section 4.2 is employed, showing a segment of the PSD calculated with smaller window sizes in figure 4.13b. Now it is easy to observe the superiority of the optimal model compared to the memoryless model.

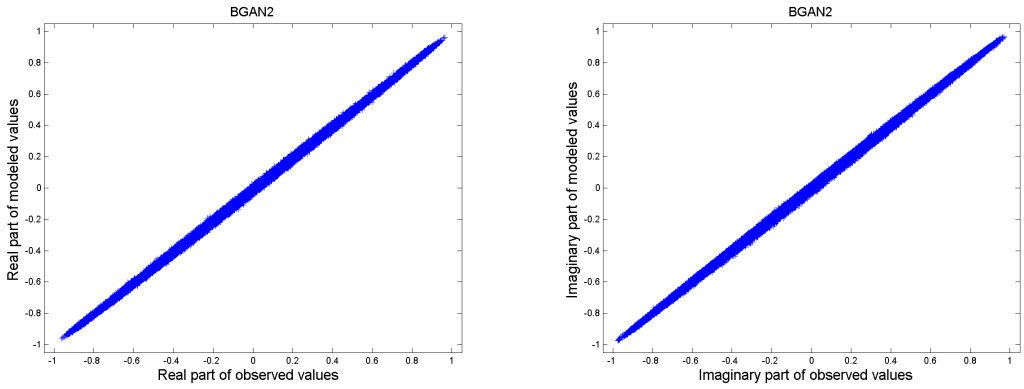
BGAN2

The parameter optimization procedure also now led to equal parameters for the empirical polynomial and the empirical spline model, and is thus presented in the same table (table 4.6). Again, the small differences in accuracy between the polynomial and the spline models enables

PA model parameters	MSE [dB]	
	Polynomial	**Spline**
Memoryless	-29.377	-29.425
$Q_1 = 1$, $Q_2 = 3$, $d^{(rf)} = 2$	-29.708	-29.769
$Q_1 = 1$, $Q_2 = 3$, $d^{(rf)} = 2$ $\rho = 0.85$,	-32.643	-32.681
$Q_1 = 1$, $Q_2 = 3$, $d^{(rf)} = 2$ $\rho = 0.85$, $M_1 = 1$, $M_2 = 8$, $d^{(bi)} = 3$	-34.012	-34.031

Table 4.6: MSE empirical models, BGAN2.

the analysis of the general empirical model, rather than analyzing the individual results. Starting at the introduction of the memory in row two, a very small improvement of about 0.35 dB is observed. The results from the Hammerstein model presented in the previous section also showed very limited possibilities concerning an accuracy improvement for the BGAN2 case in this scenario. Thus, these results were expected. The observed 0.35 dB accuracy improvement observed here was however not observed in the previous section. This can be explained by the fact that no weighting is done in the empirical models, and hence the MSE is minimized. When introducing the thermal effects an improvement of approximately 3 dB is observed, and then another 1.5 dB when also introducing the biasing effects. Hence, an accumulated 4.6 dB accuracy improvement compared to the memoryless model is observed. This is indeed a significant improvement. Depicted in figure 4.14 are the real and imaginary plots for the optimal spline model found in the study. Only the plots from the empirical spline model are presented with

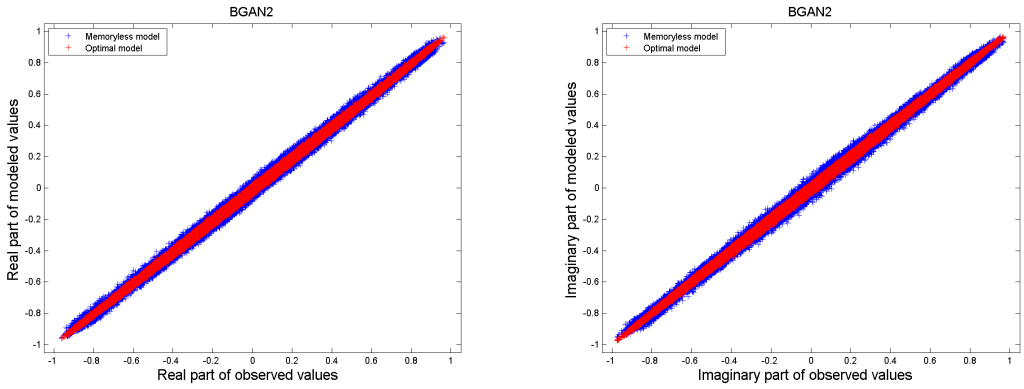


(a) Real parts.

(b) Imaginary parts.

Figure 4.14: Real and imaginary plots for the optimal empirical spline model, BGAN2.

the same reasoning as in the BGAN1 case. It is clear that the PA modeling is consistent both in amplitude and phase. Of course, the corresponding curves for the polynomial case have been inspected in the study showing the same behaviour. Figure 4.15 now plots the real and imaginary curves for the memoryless spline model along with the curves for the optimal model. The accuracy improvements obtained by introducing the thermal and biasing effects are clear from these plots. Compared to these plots in the BGAN1 case, it is now far easier to see the accuracy



(a) Real parts.

(b) Imaginary parts.

Figure 4.15: Memoryless model vs. optimal model, BGAN2.

improvements when employing the optimal empirical model. This is due to the approximately 5 dB accuracy difference in the memoryless model results for the two cases.

The quite large accuracy improvement experienced with the introduction of the two lower branches in figure 2.10 along with the lack of such when investigating the Hammerstein model in the previous section begs the question whether or not the filter modeling the RF response is at all necessary for the BGAN2 case. Shown in table 4.7 are the results obtained when omitting the filter, first employing both lower branches and then employing only the third branch in figure 2.10. These results clearly shows that practically all of the accuracy improvement experienced

PA model parameters	**MSE [dB]**	
	Polynomial	**Spline**
$\rho = 0.85$, $M_1 = 1$, $M_2 = 8$, $d^{(bi)} = 3$	-33.805	-33.816
$M_1 = 1$, $M_2 = 8$, $d^{(bi)} = 3$	-33.785	-33.793

Table 4.7: MSE empirical models, no RF filter, BGAN2.

is due to the modeling of the biasing effects. This is quite remarkable not only in terms of the difference from where the improvement stems compared to the BGAN1 case, but also in terms of the model complexity. It was made clear in section 2.6 that the number of coefficients is of no concern in this study. However, by omitting the RF filter and the thermal branch in the empirical spline model, the number of coefficients is reduced from 300 to 55 with an accuracy decrease of only 0.25 dB. The same relative numbers are found for the empirical polynomial case. This vast reduction in complexity and its implications will be further discussed in the general analysis below.

Depicted in figure 4.16 are the estimated PSDs for the BGAN2 case. The accuracy improvement when employing the optimal model compared to the memoryless model is now easily observable. The modeling of the main lobe of the spectrum is very accurate in both cases, but outside the main lobe the optimal model is clearly superior. Both models obviously struggle in the outer parts of the spectrum, but seeing as this part of the estimation suffers from some inaccuracy, this is shall not be interpreted as a big problem. Also, the dB level in this part of the PSD is low, and hence some inaccuracy here does not contribute much to the total error.

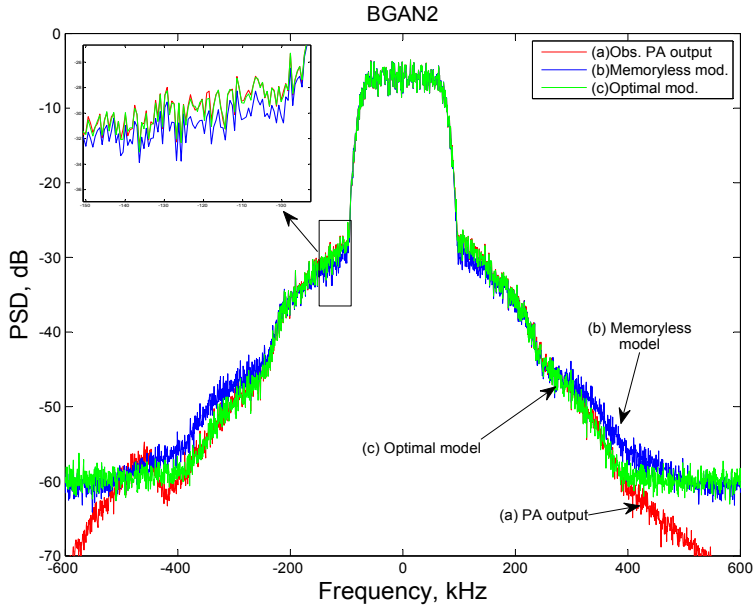


Figure 4.16: Spectrum comparison of different modeling approaches. (a) is the observed PA output, (b) the memoryless model output and (c) the optimal empirical spline model output.

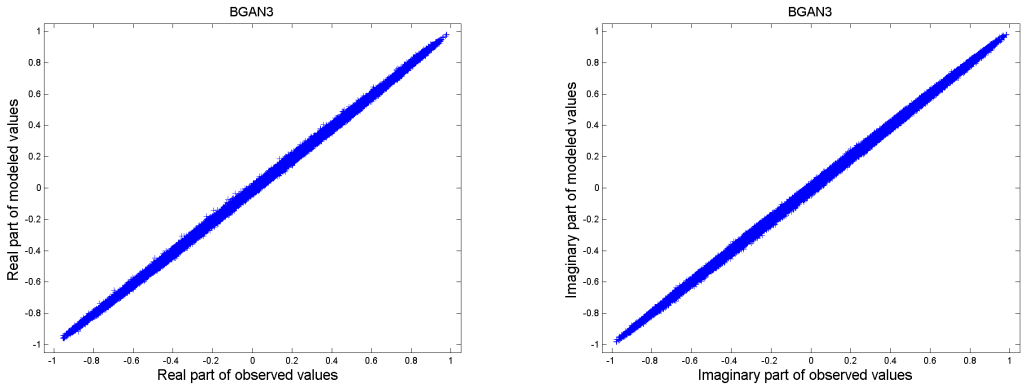
BGAN3

As for the two other cases, the parameter optimization procedure for the BGAN3 data set led to the same parameters for both the empirical polynomial and the empirical spline model. These results are presented in table 4.8. Once again the accuracy difference between the two models are

PA model parameters	**MSE [dB]**	
	Polynomial	**Spline**
Memoryless	-26.287	-26.313
$Q_1 = 1$, $Q_2 = 3$, $d^{(rf)} = 2$	-26.640	-26.671
$Q_1 = 1$, $Q_2 = 3$, $d^{(rf)} = 2$ $\rho = 0.85$,	-30.332	-30.358
$Q_1 = 1$, $Q_2 = 3$, $d^{(rf)} = 2$ $\rho = 0.85$, $M_1 = 1$, $M_2 = 8$, $d^{(bi)} = 3$	-33.625	-33.647

Table 4.8: MSE empirical models, BGAN3.

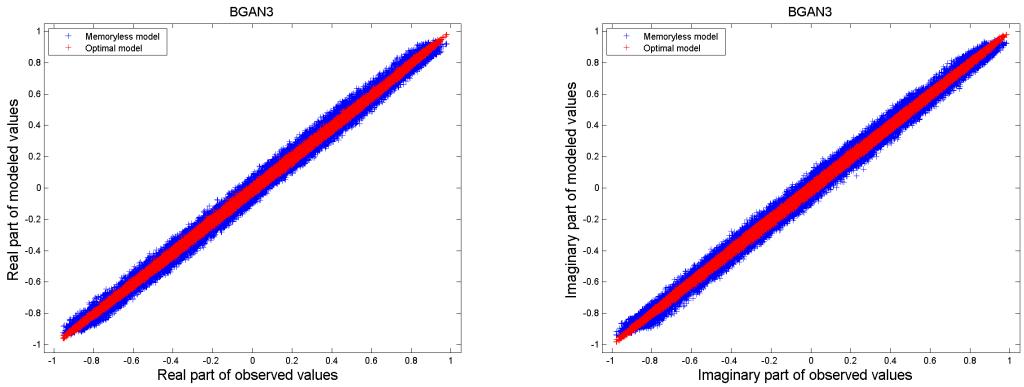
so small that an analysis of the general model is sufficient. The very limited improvement ability of the top branch experienced in the previous section is also seen here; only about 0.3 dB better performance is obtained compared to the memoryless model. The introduction of the second branch modeling the thermal effects results in an excess of 3.5 dB improvement, slightly better compared to the BGAN2 case. Completing the model, the introduction of the biasing effects in the third branch now improves the accuracy by another 3.3 dB. Thus, the biasing effects are even more important now than in the previous case. Hence, a total accuracy improvement of 7.3 dB is obtained when comparing the memoryless and the optimal models. Figure 4.17 depicts the real and imaginary plots for the optimal empirical spline model. Once again only the plots from the empirical spline model are presented as the results from the empirical polynomial model are so similar. The corresponding plots for the empirical polynomial model have of course also been inspected, showing the same relationships as the plots depicted in figure 4.17. The two plots clearly show that both the amplitude and the phase response of the PA are modeled properly. As



(a) Real parts. (b) Imaginary parts.

Figure 4.17: Real and imaginary plots for the optimal empirical spline model, BGAN3.

for the two previous cases it is interesting two plot the real and the imaginary parts of both the memoryless and the optimal models simultaneously to get a visual impression of the accuracy improvement. This has been done in figure 4.18. The 7.3 dB accuracy improvement is clearly



(a) Real parts. (b) Imaginary parts.

Figure 4.18: Memoryless model vs. optimal model, BGAN3.

seen in the figure. The variance of the optimal model compared to a straight line is far less than what the case is for the memoryless model.

When inspecting the MSE results in table 4.8 we notice the same tendencies discussed in the BGAN2 case; the modeling of the thermal and biasing effects provide for the significant improvement. Thus, the question whether or not the RF response filter in the top branch of the general empirical model depicted in figure 2.10 can be omitted, is once again relevant. Presented in table 4.9 are the results when omitting the filter, first including both thermal and biasing effects and then only biasing effects. The results are again clear; the biasing effects dominates the improvement providing for 6.9 dB alone. Noting the fact that the optimal parameters for the BGAN2 and BGAN3 case are equal, the same complexity reduction from 300 to 55 is the case for the empirical spline model presented here. The MSE increase in this case is approximately 0.5 dB, a small price to pay for such a large decrease in complexity. This complexity reduction will, as earlier stated, be discussed in the general analysis below.

PA model parameters	MSE [dB]	
	Polynomial	Spline
$\rho = 0.85$, $M_1 = 1$, $M_2 = 8$, $d^{(bi)} = 3$	-33.294	-33.308
$M_1 = 1$, $M_2 = 8$, $d^{(bi)} = 3$	-33.160	-33.169

Table 4.9: MSE empirical models, no RF filter, BGAN3.

Figure 4.19 depicts the calculated PSDs for the BGAN3 case. Only the empirical spline model results are depicted with the same reasoning concerning the empirical polynomial model as stated above. Also now the superiority of the optimal empirical model compared to the memoryless

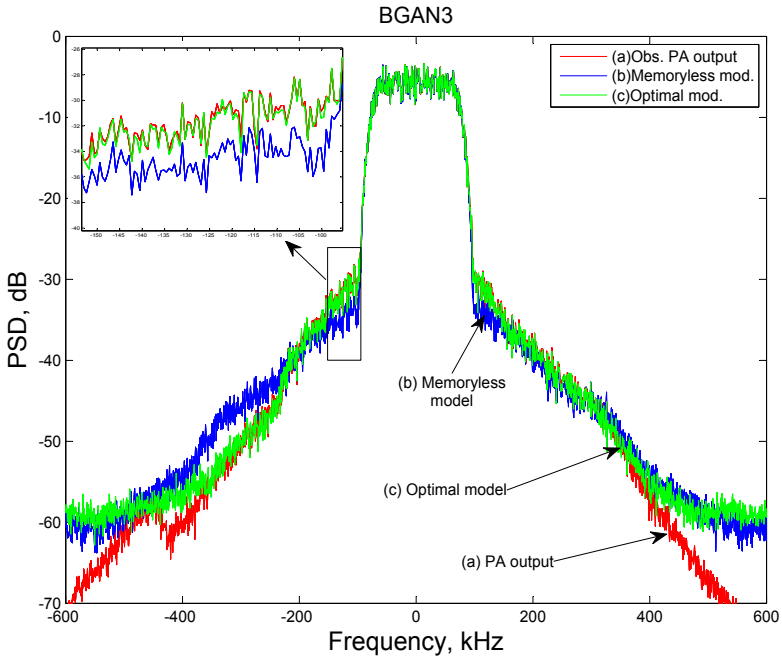


Figure 4.19: Spectrum comparison of different modeling approaches. (a) is the observed PA output, (b) the memoryless model output and (c) the optimal empirical spline model output.

model is evident. The modeling of the main lobe is very accurate in both approaches, but outside this the empirical model clearly outperforms the memoryless model. The problem in the outer parts of the frequencies are explained in the BGAN2 case above and is thus not repeated here.

General Analysis

We now further analyze and explain the results presented in this section. As presented in all three cases above, the general models dependence on whether polynomials or splines are used as basis functions in the modeling of the nonlinearity seems non existing. Both models basically perform at the same level. This is mainly due to the polynomial character of the curve (PA characteristic) which is sought to model. Figure 2.1 shows this polynomial nature. In this kind of situation, employing splines as basis functions will generally not improve the accuracy of the estimated model. If we were modeling a characteristic as the one showed in figure 2.3 (linearized PA), the situation would have been different. The polynomial approach would encounter problems in the breakpoint (at input power P_{lin}) and probably have large fluctuating errors in the area to the

right of this point. The spline approach could however place a knot at this point, experiencing no difficulties when modeling the breakpoint, and thus no negative effects in the right area of the curve. Having said this, the question why splines are still investigated as basis functions has two answers. Firstly, the memoryless and the Hammerstein model presented both employ splines as basis functions. An empirical model employing the same basis functions makes for an interesting accuracy comparison. The second reason is more complex. As described above, the polynomial nature of the PA characteristic implies that polynomial basis functions would fit nicely. However, the empirical models contain not only a single nonlinearity, every memory branch has its own nonlinearity[5]. It is not certain that each of these nonlinearities possesses the nice polynomial nature as the one presented in figure 2.1 does. In this kind of scenario, spline bases are far more flexible than polynomial bases. Employing polynomial bases may often lead to fluctuating errors in the area close to 1. This will not happen when using spline bases. Thus, spline basis functions may well outperform polynomial basis functions in the modeling of these nonlinearities, resulting in more accurate models.

The two last cases, BGAN2 and BGAN3, experienced quite large accuracy improvements when introducing the thermal and biasing effects. As pointed out in both cases the biasing effects solely provide a large part of the improvement. Above this was presented as if the thermal branch really also could be omitted, at least if complexity was an issue. However, when not only looking at the optimal cases of tables 4.6 and 4.8, it is clear that the thermal effects has a significant impact on the accuracy when the biasing branch is not part of the model. Interestingly, this impact seems to vanish when the biasing branch is introduced. This is clear when inspecting the two tables 4.7 and 4.9. This is explained by the way the coefficients of the models are estimated, outlined in section 2.6. Because of the column dependencies in the PA model, the coefficients are calculated using the SVD technique. Now, these column dependencies also mean that the different branches of the empirical models are mutually dependent. This is the reason why most of the effect of the thermal branch vanishes when the biasing branch is introduced. Had the branches been mutually independent, the MSE results in tables 4.7 and 4.9, when omitting the thermal branch, would have increased by approximately 3 and 3.5 dB (the corresponding MSE decrease when introducing the thermal branch in the two cases), respectively.

The memory overflow occurrences experienced when testing the empirical models was mentioned in section 3.2. As described, the overflow occurred at times when further accuracy improvements seemed unlikely. However, not having tested this scenario (further increasing the model parameters), one can not be sure of this. The results presented above showed that for the BGAN2 and BGAN3 cases, a considerable complexity reduction resulted in a negligible accuracy drop. This enables a further expansion of the filter in the biasing branch to investigate whether or not this would lead to improvements. However, the accuracy improvements when expanding the filter beyond the presented parameters given in table 4.7 and 4.9 were very limited. It can thus be said that the results presented in this section are the optimal ones.

The parameter selection procedure outlined in section 3.4 was designed to obtain as optimal results as possible without having to perform an exhaustive search over all parameter combinations. However, the fact that the three branches in the models are mutually dependent makes it difficult to establish the degree of optimality for the presented parameter choices. Thus, the optimality of the results must be evaluated in the context of how the parameters were chosen. One can not be sure that the optimality is general, although the expansion of the biasing filter presented above, resulting in no particular accuracy improvements, points in such a direction. Having said this, an exhaustive search over all combinations is not feasible, and the procedure employed is one way of dealing with this problem.

[5]See the top branch of figures 2.11 and 2.12.

Chapter 5

Conclusions

Modeling of power amplifiers is an important area in the field of radio communication, both when considering predistortion and system simulations. The predistortion technique enables high performance when employing bandwidth efficient modulation schemes. The inherent bandwidth limitation of the radio frequency spectrum makes this technique increasingly important as more wireless applications enter the market. In system simulations, highly accurate PA models enable the designers to better take into account the nonlinear effects of the PA in the design process. This provides a better understanding of the radio system and may be used to lower error probabilities. Starting with the presdistortion technique, four models have been presented, two of which have been investigated; the memoryless spline model and the Hammerstein model. The memoryless spline model assumes, as the name suggests, the PA characteristic to be a memoryless nonlinearity. This characteristic is then modeled using cubic splines. The Hammerstein model expands this model by inserting a filter subsequent to the memoryless nonlinearity. The filter is a linear transversal filter, modeling the memory effects of the PA. The Hammerstein model was expected to outperform the memoryless model due to the fact that a memoryless approach is clearly a simplification of reality. However, the results obtained did not unanimously point in this direction. Only in one of the three PAs modeled (the class A PA) did the Hammerstein model show a clear performance improvement. The procedure employed in the coefficient estimation of the Hammerstein model does not ensure optimality, but as made clear in the analysis, the results are optimal with no reasonable doubt. Hence it can be concluded that the Hammerstein model may perform better than the memoryless model, but this superiority is by no means general. The structure of the Hammerstein model is simply not fit to model the memory effects in all PAs.

For the case of system-level modeling, two empirical models have been investigated. The models consist of three branches, the first one modeling memory effects as in the Hammerstein model, the second one modeling thermal effects and the third one modeling biasing effects. The two models presented differ only in the basis functions used to model the memoryless nonlinearities (polynomials vs splines). The models obtained very similar results. This is, as described in the previous chapter, due to the fact that both the polynomial and the spline basis functions model the nonlinearities nicely. When comparing the empirical models to the memoryless and Hammerstein models, the accuracy improvement is clear. The modeling of the thermal and biasing effects had a significant impact on the accuracy. The importance of these, the biasing effects in particular, increased when moving from a class A to a class AB and finally to a class B PA. This was expected, seeing that class B PAs are more exposed to memory effects than class AB PAs, which again are more exposed than class A PAs.

When inspecting the results of the empirical models more closely, it is clear that the accuracy

improvements obtained stems from different sources. The top branch of the models provided for most of the improvement for the class A PA. Remembering the results from the Hammerstein model, this improvement was expected. The thermal and biasing effects played less important roles in this case. The scenario for the class AB and class B PAs was the complete opposite; the thermal and biasing effects provided for almost the entire improvement. Thus, the final structure of the optimal empirical model is not identical in all three cases. Hence it can be concluded that the structure of the optimal model is by no means general. This problem is also noted in [18], where it is made clear that the accuracy of empirical models is highly sensitive to the structure of the model and the parameter extraction procedure. A distinct model structure optimal for one particular data set is by no means obvious to perform well for a different data set. This is exactly what has been experienced in this study. The optimal structure for the class AB and class B PAs are equal, but these data sets are drawn from the same PA, operated at different classes, so this was no big surprise. However, the optimal structure of the class A PA is different. Generally, the empirical models obtained very accurate results for all three PAs modeled in the study. Hence, it can be concluded that the empirical models obtain significant accuracy improvements, but they still suffer from doubtful generality.

The empirical models in this study are based on the work presented in [4], and a result comparison is thus interesting. The class B PA is modeled in both studies, and is thus the best basis for a comparison. The expansion of the RF and biasing filters from causal to non-causal filters provided very limited improvements. Actually, if also aiming at minimizing the number of coefficients in this study, both filters would have been chosen causal. The introduction of splines as basis functions did not improve the model accuracy, as thoroughly described in the previous chapter. As for the altering of the thermal filter from a causal low-pass filter to a recursive filter, a comparison was at first impossible to make as [4] provided no results for the employment of the thermal effects, but no biasing effects. However, after consulting the author of [4], Nima Safari, such results were provided; an MSE of -32.1 dB when employing a three-tap RF filter and a four-tap thermal filter, but no compensation for biasing effects. Both filters are causal and uses the sparse delay-tap structure presented in section 2.6 with parameters 2 and 3 for the RF and biasing filters, respectively. Comparing that result to what was obtained in this study, the low-pass approach performs better. However, when only comparing this one case, concluding that the low-pass approach is superior on a general basis can not be done. Generally speaking, the optimal results are very much similar, and it is thus hard to prefer one model to another. The great accuracy improvement caused by the biasing effects, along with the limited effect of expanding the filter in this branch from causal to non-causal, makes it hard to conclude whether or not the models presented here would perform better than the model presented in [4] on a general basis. As mentioned above, empirical models suffers from doubtful generality, and it is thus difficult to conclude whether or not one model outperforms another on a general basis.

The empirical models presented in the study were developed to perform well both for narrowband and wideband applications. Unfortunately all data sets available for testing of the models were narrowband, and so, the applicability concerning wideband applications has not been investigated. However, the results obtained in this study coincides well with the results presented in [4], the work which was the basis of the empirical models presented here. [4] also presents large accuracy improvements for wideband applications. Thus, if the empirical models presented in this study were to be tested with a wideband data set, large accuracy improvements would have been expected.

5.1 Further Work

The models and methods presented in this study leaves lots of room for further work. Presented in the following are some of the most relevant research topics for further investigation.

- The estimation procedure for the coefficients in the Hammerstein model could well use some more work. If one is able to establish whether or not the error function in the iterative procedure is strictly convex, much time and work can be saved. This is however not a trivial problem, and different PAs will most probably give different answers to the question. Armed with a simple method to test for strict convexity, the applicability of the coefficient estimation procedure presented here could be determined with certainty.

- In section 2.2 on predistortion three models were presented, but only one of them (the Hammerstein model) was investigated. As for the two others, the Wiener- and the Wiener-Hammerstein models, these represent alternative ways to model the memory effects. The reason why these were not investigated is the mathematical complexity in the coefficient estimation. More specifically in this study; the complexity of the calculation of the Jacobian matrices. Hence, coefficient estimation procedures for these two models is an area of further work. The SVD technique is an obvious alternative. However, the desirable weighting described in section 3.3 is a problem in this context. Other iterative procedures not requiring a user-supplied Jacobian matrix is another possibility for the coefficient estimation.

- Inspired by the work done in [4], it is of course possible to expand the optimality objective of the empirical models also to minimize the number of coefficients. A small number of coefficients obviously leads to a less complex model, which is often desirable.

- The empirical models leave a lot of room for further work due to their many degrees of freedom. First of all, the parameter selection procedure might well lead to better results if done otherwise. As mentioned several times, an exhaustive search over all possible parameter combinations is not feasible, but such a search over a selected subset of the possibilities could be done. An expansion of the general model, adding new branches for other known effects, is also a possibility for further work.

- Proper testing for the wideband case is very important. This will establish with certainty the empirical models' ability to work well both for narrowband and wideband applications.

- Adaptive predistortion. Here the whole predistortion process would be able to adapt to changes in the working environments. This means a continuous modeling of the PA, along with a predistorter being able to adapt to changes in the PA model. This could, if implemented properly, improve the predistortion operation considerably and thus also the overall performance of the radio communication system.

Bibliography

[1] N. Safari, P. Fedorenko, J. Stevenson Kenney, and T. Røste. Spline-based model for digital predistortion of wide-band signals for high power amplifier linearization. *IEEE/MTT-S international microwave symposium*, pages 1441–1444, June 2007. 1, 2.1, 2.2, 2.2

[2] Øyvind K. Lensjø. Nonlinear modeling of power amplifiers by means of splines, 2007. Project Thesis. 1.2, 2.5

[3] N. Safari, N. Holte, and T. Røste. Digital predistortion of power amplifiers based on spline approximations of the amplifier characteristics. *IEEE vehicular technology conference*, pages 2075–2080, October 2007. 1.2, 2.1, 2.4, 4.1

[4] Nima Safari. *Linearization and Efficiency Enhancement of Power Amplifiers Using Digital Predistortion*. PhD thesis, Division of Electronics and Telecommunications, NTNU, 2008. 1.2, 2.3, 2.6, 16, 2.6.1, 18, 19, 1, 4.3, 5, 5.1

[5] L. Ding, R. Raich, and G. Tong Zhou. A hammerstein predistortion linearization design based on the indirect learning architecture. *IEEE international conference on acoustics, speech and signal processing*, 3:2689–2692, May 2002. 2.1, 2.2

[6] J. R. Barry, E. A. Lee, and D. G. Messerschmitt. *Digital Communication*. Springer, New York, third edition, 2004. 1, 2

[7] Gunnar Stette. *Radiokommunikasjon*. Tapir akademiske forlag, Trondheim, 2006. 2.1, 2

[8] Tor A. Ramstad. *Representing Information by Signals. An Introduction,* 5^{th} *revision*. Division of Electronics and Telecommunications, NTNU, Trondheim, December 2005. 2.1

[9] John G. Proakis and Dimitris G. Manolakis. *Digital Signal Processing. Principles, algorithms and applications*. Prentice Hall, New Jersey, third edition, 1996. 5, 3.5

[10] Børge E. Kristiansen. Higher order modulation and nonlinearity, 2007. Project Thesis. 6

[11] P. Lancaster and K. Salkauskas. *Curve and surface fitting: an introduction*. Academic Press, London, 1986. 2.4, 4.1

[12] Gilbert Strang. *Linear Algebra and its Applications*. Thomson Brooks/Cole, Belmont CA, fourth edition, 2006. 2.4, 2.6.1, 20

[13] T.F. Coleman and Y. Li. An interior, trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6:418–445, 1996. 2.5

[14] T.F. Coleman and Y. Li. On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds. *Mathematical Programming*, 67(2):189–224, 1994. 2.5

[15] John L. Troutman. *Variational Calculus and Optimal Control, Optimization with Elementary Convexity*. Springer, New York, second edition, 1996. 13

[16] D. R. Morgan, Z. Ma, J. Kim, M. G. Zierd, and J. Pastalan. A generalized memory polynomial model for digital predistortion of RF amplifiers. *IEEE Trans. Signal Processing*, 54(10):3852–3860, October 2006. 2.6.1

[17] Simon Haykin. *Adaptive Filter Theory*. Prentice Hall, New Jersey, fourth edition, 2002. 3.3.2

[18] J.C. Pedro and S.A. Maas. A comparative overview of microwave and wireless power-amplifier behavioral modeling approaches. *IEEE Trans. Micro. Theory. Tech.*, 53(4):1150–1163, April 2005. 5