JAMES COOK UNIVERSITY
AUSTRALIA

Adaptive Digital Predistortion For Wideband High Crest Factor Applications Based On The WACP Optimization Objective

Thesis submitted by
Bradley Dean LAKI, BE (Hons) QUT, MSc UA
in July 2013

for the degree of Doctor of Philosophy
in the School of Engineering & Physical Sciences
James Cook University

Statement On The Contribution Of Others

The following people deserve special mention for their contribution to this research. Principal supervisor Cornelis Jan Kikkert for his research advice, laboratory provision and manuscript proof reading. Workshop manager John Renehan for his supply of electrical test equipment and components. Radio technicians David Henry and Rod Prior (Broadcast Australia), Gary Lane (LaneComm Communications) and Ian Smart (Channel 7) for their transmitter site access and data collection efforts. Finally, Jasper Taylor and UniQuest for their assistance in patent filing and commercialization. This research was made possible via a Commonwealth APA scholarship.

Acknowledgments

I wish to thank the following people for their support throughout this journey. First and foremost my family for their unconditional love. My supervisors Cornelis Jan Kikkert and Owen Kenny for their guidance. Jeff Dale and Graham Weinberg for their humor and down to earth nature. Craig and Andrea McPherson for too many things to list here. Thanks guys! Phil Turner for giving me the opportunity to lecture and last but not least, Greg Borger from Ergon Energy for his support whilst preparing this final manuscript.

Abstract

Modern digital communication systems utilize OFDM and DS-CDMA signals because of their superior spectral efficiency and multiple access properties respectively. Examples of such systems include digital television (DVB-T), digital radio (DAB) and 3rd Generation mobile (WCDMA). A downside to using these signals however is the need to accommodate their very high Crest Factor, also known as Peak-to-Average Power Ratio, during transmission. Compared to other communication signals transmitted with the same average power, these signals generate greater transmitter distortion since their larger signal peaks drive the transmitter power amplifier into regions of greater nonlinearity.

Reducing this Crest Factor related distortion, whilst concurrently maintaining power amplifier efficiency, requires the power amplifier's transfer characteristic to be externally linearized. For OFDM and DS-CDMA signals, *digital predistortion* is the favored linearization technique given its cost effectiveness, superior signal processing capability, potential to adapt and ability to linearize the entire transmitter.

In this research thesis, a new *digital predistortion* technique is proposed. By employing *frequency-domain information feedback*, rather than traditional *time-domain signal feedback*, the new technique avoids bandwidth limitations in the feedback path and is hence more suited to wideband applications than current generation techniques. This *frequency-domain information feedback* is based on the novel *Weighted Adjacent Channel Power (WACP)* linearization objective. By incorporating frequency dependent weighting into the standard accumulation of Adjacent Channel Power (ACP), this novel linearization objective is able to discriminate between spectral distortion components and hence control the location of spectral distortion reduction. This makes linearization more robust in the presence of residual predistortion filtering inaccuracies.

To accommodate the power amplifier memory effects associated with wideband signal modulation, the new technique uses a predistortion filter architecture based on the classic nonlinear dynamic Volterra Series. Mindful of Volterra kernel efficiency, the technique further applies a novel hybridized triple-stage pruning strategy, leaving the kernel size not only linear with respect to memory, but also independent of

hardware sampling rate implementation. This pruning activity ultimately reduces the number of dynamic filter parameters needing to be estimated.

This new *digital predistortion* technique utilizes generic mathematical optimization to estimate the resulting predistortion filter kernel, with frequency-domain WACP interpreted as the linearizing optimization objective. In line with theory, the objective is assumed to be nonconvex and hence both global and local optimization algorithms are employed to achieve true global convergence. This is in direct contrast to the traditional and competing *Direct Learning* technique which is more focused on quick local convergence, and less focused on guaranteeing maximum linearization performance corresponding to the objective global minimum.

Since predistortion filter kernel estimation is performed using the transmitted wideband signal, the new technique is ultimately on-air adaptive. This means any transmitter using the technique will be both on-air and optimally linearized for its entire operational life.

Contents

List of Tables

List of Figures

Acronyms

3G 3rd Generation

ABB Alpha Branch & Bound

ABC Australian Broadcasting Corporation

ACD Adjacent Channel Distortion

ACMA Australian Communications and Media Authority

ACP Adjacent Channel Power

ACPR Adjacent Channel Power Ratio

ADC Analog-to-Digital Converter

a.k.a. Also Known As

AM-AM Amplitude Modulation to Amplitude Modulation

AM-PM Amplitude Modulation to Phase Modulation

BER Bit Error Rate

BW Bandwidth

CCD Co-Channel Distortion

CCDF Complementary Cumulative Distribution Function

CCP Co-Channel Power

CCPR Co-Channel Power Ratio

CDMA Code Division Multiple Access

CF Crest Factor or Center Frequency

CM Counter-Measure

CPM Continuous Phase Modulation

DAB Digital Audio Broadcasting

DAC Digital-to-Analog Converter

dB deciBel

dBc deciBel relative to Carrier

dBm deciBel relative to milli Watt

DDR Dynamic Deviation Reduction

DOS Disk Operating System

DS-CDMA Direct Sequence Code Division Multiple Access

DSP Digital Signal Processor

DVB-T Digital Video Broadcasting - Terrestrial

EE & R Envelope Elimination & Restoration

EIRP Effective Isotropic Radiated Power

ETSI European Telecommunications Standards Institute

EVM Error Vector Magnitude

FCC Federal Communications Commission

FET Field Effect Transistor

FIR Finite Impulse Response

FM Frequency Modulation

FPGA Field Programmable Gate Array

GaAs Gallium Arsenide

GD Gradient Descent

GPIB General Purpose Interface Bus

Hz Hertz

IDE Integrated Development Environment

IEEE Institute of Electrical and Electronics Engineers

IF Intermediate Frequency

IFFT Inverse Fast Fourier Transform

IIR Infinite Impulse Response

IMT-2000 International Mobile Telecommunications - 2000

IP Intellectual Property

IQ Inphase Quadrature

KW Kilo Watt

LAC Lower Adjacent Channel

LDMOS Laterally Diffused Metal Oxide Semiconductor

LINC LInear amplification using Nonlinear Components

LMS Least Mean Squares

LTI Linear Time Invariant

LU Lower Upper

LUT Look Up Table

MER Modulation Error Ratio

MHz Mega Hertz

MSK Minimum Shift Keying

NEC Nippon Electric Company

NMA Nonlinear Moving Average

NMS Nelder-Mead Simplex

OBO Output Back Off

OFDM Orthogonal Frequency Division Multiplexing

PA Power Amplifier

PAPR Peak-to-Average Power Ratio

PC Personal Computer

PDF Probability Density Function

PSD Power Spectral Density

PSK Phase Shift Keying

QAM Quadrature Amplitude Modulation

QPSK Quadrature Phase Shift Keying

RAM Random Access Memory

RBW Resolution BandWidth

R&D Research & Development

RF Radio Frequency

RLS Recursive Least Squares

RRC Root Raised Cosine

SBS Special Broadcasting Service

SPFL Spectral Power Feedback Learning

SR1 Symmetric-Rank-1

SWT SWeep Time

TM-I Transmission Mode - I

TNTB Twin Nonlinear Two-Box

TRN Trust Region Newton

TWT Travelling Wave Tube

UAC Upper Adjacent Channel

UHF Ultra High Frequency

UMTS Universal Mobile Telecommunications System

V Volt

VBW Volterra Behavioral Wideband or Video BandWidth

vs Verses

W Watt

WACP Weighted Adjacent Channel Power

WCDMA Wideband Code Division Multiple Access

Mathematical Symbols

N.B. Vector parameters are represented by boldface text e.g. $\boldsymbol{h}$

$\triangleq$ Defined as

Δ Trust region radius (TRN algorithm)

Δf Discrete frequency step size

$\|\cdot\|$ Euclidean Norm

$\lceil\cdot\rceil$ Ceiling operator

$[\cdot]^T$ Vector transposition

α_i Variable (i^{th} dimension) ensuring convexity of $\mathcal{L}(\boldsymbol{h})$ (ABB algorithm)

$\alpha_i^{\textbf{ideal}}$ Ideal α_i (ABB algorithm)

$\alpha_i^{\textbf{prac}}$ Practically computable α_i (ABB algorithm)

χ Chromosomes used to estimate initial Population Pool (Genetic algorithm)

ϵ Small positive scalar used in Finite Difference estimation

κ Number of chromosomes in Mating Pool (Genetic algorithm)

λ General eigenvalue of $\boldsymbol{H}$

$\boldsymbol{\Lambda}$ Diagonal spectral matrix of $\boldsymbol{H} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$

μ Step distance (GD algorithm)

$\mu^{\textbf{ideal}}$ Ideal step distance (GD algorithm)

$\mu_{\boldsymbol{h}_k}$ Step distance chosen for iteration $\boldsymbol{h}_k$ (GD algorithm)

$\boldsymbol{\Phi}$ Diagonal matrix of α values (ABB algorithm)

σ Scalar parameter of the Symmetric-Rank-1 Update equation

Υ Mean square criterion on which to automate exiting (NMS algorithm)

Υ_0 Exit threshold for Υ (NMS algorithm)

ϱ Number of chromosomes in Offspring Pool (Genetic algorithm)

ϑ Chromosome concentrations when refinement is halted (Genetic algorithm)

ξ Number of chromosomes in Population Pool (Genetic algorithm)

$\boldsymbol{A}[\,\cdot\,]$ System amplifier operator within baseband transmitter model

$\hat{\boldsymbol{A}}[\,\cdot\,]$ System amplifier operator within RF transmitter model

$\boldsymbol{A}_n[\,\cdot\,]$ n^{th} order amplifier operator within baseband transmitter model

$\boldsymbol{A}_n\big\{\, s[k], \ldots, s[k] \,\big\}$ n-linear amplifier operator within baseband transmitter model

$\hat{\boldsymbol{A}}_n[\,\cdot\,]$ n^{th} order amplifier operator within RF transmitter model

$\hat{\boldsymbol{A}}_n\big\{\, \tilde{s}_1(t), \ldots, \tilde{s}_n(t) \,\big\}$ n-linear amplifier operator within RF transmitter model

$\boldsymbol{a}$ Linear parameter of linear parametric model (Least Squares $\boldsymbol{g}$ estimation)

$a_n(\tau_1, \ldots, \tau_n)$ n^{th} order amplifier Volterra kernel within baseband transmitter model

$\hat{a}_n(\tau_1, \ldots, \tau_n)$ n^{th} order amplifier Volterra kernel within RF transmitter model

$B(\boldsymbol{h})$ Optimization objective as a function of $\boldsymbol{h}$

b Value of objective function. $B(\boldsymbol{h}) = b$

b_o Value of objective function at optimization algorithm's current iterate

b_{min} Global minimum value of objective function. $B(\boldsymbol{h}_o) = b_{min}$

B Spectral bandwidth of $s(t)$ (Hertz)

B_c Continuous-time spectral bandwidth (Hertz)

B_d Discrete-time spectral bandwidth (cycles/sample). $B_d = B_c/F_s$

$\boldsymbol{\mathcal{C}}$ Infinite set representing the complex number plane

C Scalar averaging term (Least Squares $\boldsymbol{g}$ estimation)

c Constant parameter of linear parametric model (Least Squares $\boldsymbol{g}$ estimation)

$c^i_{\boldsymbol{h}_k}$ i^{th} candidate step distance generated during iteration $\boldsymbol{h}_k$ (GD algorithm)

D_i i^{th} Gerschgorin disk (ABB algorithm)

d Small positive increment used to define S_0 (NMS algorithm)

$\boldsymbol{d}$ Deviation from optimization algorithm's current iterate

$\hat{\boldsymbol{d}}$ Modified $\boldsymbol{d}$ vector (Least Squares $\boldsymbol{g}$ estimation)

$\boldsymbol{e}_i$ Unit vector along the i^{th} dimension of the vector space $\boldsymbol{h}$

f Frequency (Hertz)

$f_{\textbf{Carrier}}$ Carrier frequency

f_E Transmission band edge frequency (Hertz)

F_s Sampling frequency (samples/second)

f_0 Adjacent channel frequency at which the weighting drops to zero. $W(f_0) = 0$

G Amplifier gain

$\boldsymbol{g}$ Gradient vector

$\boldsymbol{H}$ Hessian matrix

$\boldsymbol{H}[\,\cdot\,]$ General system Volterra operator

$\boldsymbol{H}_n[\,\cdot\,]$ General n^{th} order Volterra operator

$\boldsymbol{H}_n\{s_1(t),\ldots,s_n(t)\}$ General n-linear Volterra operator

$H_n(f_1,\ldots,f_n)$ General n-dimensional Fourier Transform of $h_n(\tau_1,\ldots,\tau_n)$

$\boldsymbol{H}_B(\boldsymbol{h})$ Hessian matrix of $B(\boldsymbol{h})$

$[\boldsymbol{H}_B]^{R_g}$ R_g's objective function interval Hessian matrix (ABB algorithm)

$\underline{\boldsymbol{H}}$ Lower interval bound of $[\boldsymbol{H}_B]^{R_g}$

$\bar{\boldsymbol{H}}$ Upper interval bound of $[\boldsymbol{H}_B]^{R_g}$

$\boldsymbol{H}_{\mathcal{L}}(\boldsymbol{h})$ Hessian matrix of $\mathcal{L}(\boldsymbol{h})$

$h_n(\tau_1,\ldots,\tau_n)$ General n^{th} order Volterra kernel

$\boldsymbol{h}$ Optimization vector space (row vectorized)

$\boldsymbol{h}^L$ Lower constraint on the optimization vector space $\boldsymbol{h}$

$\boldsymbol{h}^U$ Upper constraint on the optimization vector space $\boldsymbol{h}$

$\boldsymbol{h}^{L,R_g}$ Lower constraint on the general boxed region R_g

$\boldsymbol{h}^{U,R_g}$ Upper constraint on the general boxed region R_g

$\boldsymbol{h}_k$ k^{th} iteration of the optimization algorithm

$\boldsymbol{h}_o$ Optimal value of $\boldsymbol{h}$ which minimizes the objective function. $B(\boldsymbol{h}_o) = b_{min}$

$\boldsymbol{h}^s$ Vector space starting point of mathematical optimization (NMS algorithm)

$\boldsymbol{h}_s$ Scheduled optimization variable vector. $\boldsymbol{h}_s \subset \boldsymbol{h}$

$\boldsymbol{I}$ Identity matrix

$i_1, \ldots, i_m$ Delay variables as used in $p_m[i_1, \ldots, i_m]$

i Delay variable as used in $p_m[i]$ and M estimation distortion probing

L Maximum optimization load

LAC Integration domain of the Lower Adjacent Channel frequencies

$\mathcal{L}(\boldsymbol{h})$ Convex function underestimating $B(\boldsymbol{h})$ over region R_g (ABB algorithm)

$\boldsymbol{M}[\,\cdot\,]$ Mask filter operator within RF and baseband transmitter models

M Predistortion filter discrete-time memory length (samples)

M_c Predistorter continuous-time memory length (seconds). $M_c = MT_s$

$m(\,\boldsymbol{d}\,)$ Objective function quadratic model (TRN algorithm)

n Size of the optimization variable vector $\boldsymbol{h}_s$. $n = S$ for the special case $\boldsymbol{h}_s = \boldsymbol{h}$

P Maximum nonlinearity (odd) of predistortion filter

$\boldsymbol{P}$ Vector averaging term (Least Squares $\boldsymbol{g}$ estimation)

$\boldsymbol{P}\big[\,\cdot\,\big]$ System predistortion filter operator

$\boldsymbol{P}_m\big[\,\cdot\,\big]$ m^{th} order predistortion filter operator

$p_m[i_1, \ldots, i_m]$ m^{th} order predistorter Baseband Volterra kernel

$p_{2a+1}[i_1, \ldots, i_{2a+1}]$ Kernel of compact predistorter representation

$p_{2a+1}[i_1, \ldots, i_a]$ Predistorter kernel after *Volterra Behavioral Wideband* pruning

$p_{2a+1}[i]$ Predistorter kernel after *Dynamic Deviation Reduction* pruning

$\tilde{p}_{2a+1}[i]$ Predistorter kernel after *R-Sample Delay Increment* pruning

$p_m[i]$ m^{th} order predistorter kernel after hybrid pruning

$\boldsymbol{p}_m$ Set of m^{th} order predistorter kernel coefficients (row vectorized)

$\boldsymbol{p}_{mR}$ Real part of $\boldsymbol{p}_m$

$\boldsymbol{p}_{mI}$ Imaginary part of $\boldsymbol{p}_m$

$P(f)$ Power spectral density as a function of frequency (Watts/Hertz)

$\boldsymbol{q}$ General eigenvector of $\boldsymbol{H}$

$q(t)$ Output of predistorter-amplifier or equivalent cascade

$q_l(t)$ Output of l^{th} order equivalent cascade nonlinearity

$q[k]$ Discrete-time version of $q(t)$

$q_l[k]$ Discrete-time version of $q_l(t)$

$\boldsymbol{Q}$ Orthogonal modal matrix of $\boldsymbol{H} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$

$\boldsymbol{Q}[\,\cdot\,]$ System operator of equivalent cascade

$\boldsymbol{Q}_l[\,\cdot\,]$ l^{th} order operator of equivalent cascade

R Delay increment within predistorter pruned Volterra Series (samples)

$\boldsymbol{R}$ Matrix averaging term (Least Squares $\boldsymbol{g}$ estimation)

$\boldsymbol{\mathcal{R}}$ Infinite set representing the real number line

R_g General boxed region of optimization vector space $\left[\boldsymbol{h}^{L,R_g} \leq \boldsymbol{h} \leq \boldsymbol{h}^{U,R_g}\right]$

S Size of the optimization vector space $\boldsymbol{h}$

S_0 Initial simplex (NMS algorithm)

S_k Geometrically updated simplex at optimization step $k \geq 1$ (NMS algorithm)

$\mathcal{S}_{ss}(f)$ Power spectral density of $s(t)$

$\mathcal{S}_{q_l q_l}(f)$ Power spectral density of $q_l(t)$

$s(t)$ Signal modulator output signal within RF and baseband transmitter models

$s[k]$ Discrete-time version of $s(t)$

$\tilde{s}(t)$ RF excitation signal within RF transmitter model

$\tilde{s}_{+1}(t)$ Positive bandpass frequency component of $\tilde{s}(t)$

$\tilde{s}_{-1}(t)$ Negative bandpass frequency component of $\tilde{s}(t)$

T_s Sampling period (seconds/sample)

UAC Integration domain of the Upper Adjacent Channel frequencies

$\boldsymbol{U}_l\big[\, s[k]\,\big]$ Accumulation of l^{th} order parasitic distortion components

$\boldsymbol{u}$ Unit vector representing the line of $B(\boldsymbol{h})$ steepest descent (GD algorithm)

$\boldsymbol{v}$ Vector parameter of the Symmetric-Rank-1 Update equation

$\bar{\boldsymbol{v}}$ Centroid of S_{k-1}'s best n vertices (NMS algorithm)

$\boldsymbol{v}_1 \rightarrow \boldsymbol{v}_{n+1}$ Ordered vertices of the previous simplex S_{k-1} (NMS algorithm)

$\boldsymbol{v}_e$ Simplex expansion point (NMS algorithm)

$\boldsymbol{v}_{ic}$ Simplex inside contraction point (NMS algorithm)

$\boldsymbol{v}_{oc}$ Simplex outside contraction point (NMS algorithm)

$\boldsymbol{v}_r$ Simplex reflection point (NMS algorithm)

$W(f)$ Nonnegative weighting as a function of frequency

W_E Weighting at the transmission band edge frequency f_E

$\boldsymbol{w}$ Parameter estimation vector (Least Squares $\boldsymbol{g}$ estimation)

$Y_n(f)$ Resultant n^{th} order spectral regrowth

$Y_n(f_1, \ldots, f_n)$ Component n^{th} order spectral regrowth

Chapter 1

Introduction

Modern digital communication systems which utilize nonconstant envelope, high Crest Factor signal modulations such as OFDM and DS-CDMA, must linearize their power amplifiers in order to achieve power efficient transmission. Of all linearization techniques, digital predistortion is considered the most promising given its cost effectiveness and superior signal processing capability.

Predistortion linearization is achieved by inserting a nonlinear digital filter at the output of the signal modulator and tuning its transfer characteristic to be the inverse of all downstream transmitter nonlinearities. As a result, the design of any digital predistortion system can be framed by two fundamental questions. Firstly, what architecture will the predistortion filter take? Secondly, how will the characterizing parameters of this filter be estimated? In line with these two design questions, this thesis presents a digital predistortion technique, novel in two primary respects:

The predistortion filter architecture is based on the dynamic Volterra Series, judiciously pruned to achieve a kernel size linear with respect to memory, without adversely affecting linearization potential. This is in direct contrast to traditional dynamic filter architectures whose kernel size remains exponential with respect to memory. This change drastically reduces the number of dynamic filter parameters needing to be estimated, making it far more robust and suited to wideband applications than current filter architectures.

The predistortion filter parameter estimation strategy is underpinned by frequency-domain information feedback and generic single objective mathematical optimization. This is in direct contrast to traditional predistortion techniques which utilize time-domain signal feedback and linear regression optimization. This change eliminates bandwidth limitations in the feedback path and ensures global optimization convergence, ultimately making the technique far more robust and suited to wideband applications than current generation techniques.

Figure 1.1 summarizes the proposed digital predistortion technique, highlighting its primary wideband novelties in yellow and green respectively. Standard transmitter elements are represented in blue.

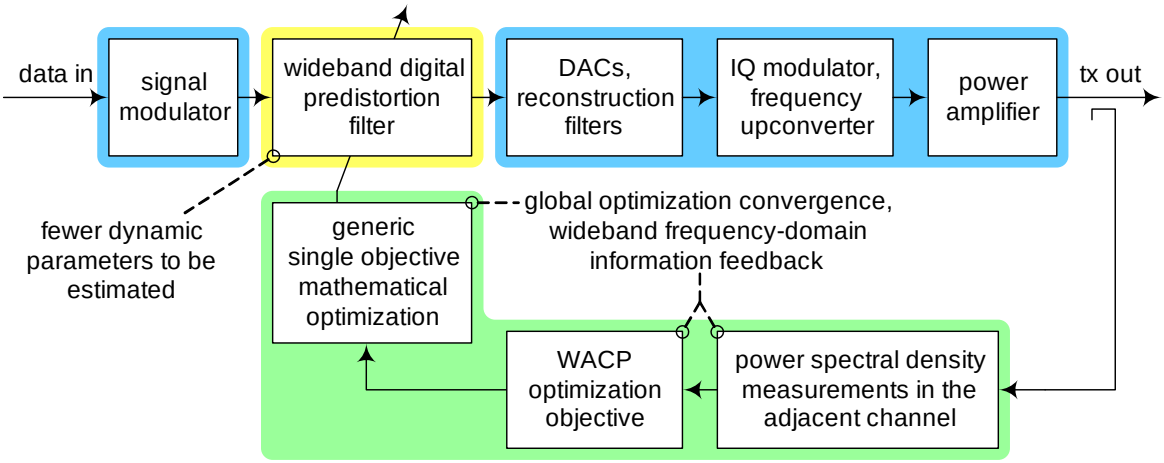


Figure 1.1: Summary of proposed digital predistortion technique

It is seen here that predistortion filter parameter estimation is modeled as a generic single objective mathematical optimization problem. That is, the predistortion filter's characterizing parameters are interpreted as a set of variables $\boldsymbol{h}$ to be optimized and a new frequency-domain measure of transmitter output nonlinearity, referred to as *Weighted Adjacent Channel Power (WACP)*, is interpreted as a dependent objective function of those variables $B(\boldsymbol{h})$. The goal is to find the optimal predistortion filter parameters which minimize the frequency-domain measure of transmitter output nonlinearity and therefore linearize the transmitter cascade. This optimization model assumes a nonconvex objective function and therefore utilizes both global and local optimization algorithms to achieve true global convergence.

In the process of developing these primary wideband novelties, four additional supporting novelties are developed. These relate to:

- the definition of the *Weighted Adjacent Channel Power (WACP)* optimization objective (as introduced above). In addition to conveying complete adjacent channel spectral regrowth behavior, this objective is able to discriminate between spectral distortion components and hence control the location of spectral distortion reduction.

- the demonstration of a new experimental procedure for estimating predistortion filter memory. This procedure is based on sweep-probing the transmitter with memory specific distortion created by the predistortion filter, and looking for changes in the signature of the output adjacent channel power spectrum.

- the derivation of predistortion filter parameter optimization schedules, both *Initial Setting* and *On-Air Adaption*, based on the concept of *influential subsets*. This concept ensures optimization is always performed over the *minimally sized variable vector which guarantees complete observability* and hence optimization convergence reliability is always maximal.

- the development of the *Distortion Array*; a graphical organizing tool for keeping track of nonlinear distortion components generated by the predistorter-amplifier cascade. Using this tool, all facets of the predistortion process can be described both visually and intuitively in the time-domain.

The original contributions of this work are summarized in two full length journal articles of the IEEE Transactions on Broadcasting and further authenticated by the application of an international patent in 2011 and successful international search report in 2012. These research outputs are discussed further in Chapter 3.

The contents of this thesis are spread across 15 chapters. In Chapters 1–3, the scope of research is set with an introduction to the transmitter linearization problem, a review of the literature and a specifically defined statement of research. In Chapter 4, the laboratory transmitter testbed is presented, highlighting the relevance of real hardware testing with DVB-T, WCDMA and DAB signal modulation. In Chapters 5–7, underlying predistortion theory is developed in preparation for Chapters 8–12, where the main body of research takes place. In Chapters 13–14, the new technique's performance is baselined and tested, with results demonstrating superiority over current generation techniques. In the concluding Chapter 15, research contributions are summarized and the future direction of this research is discussed.

With research context now established, attention is turned to formally introducing the engineering problem and developing the proposed wideband digital predistortion technique.

1.1 The Need For Transmitter Linearization

A conventional radio transmitter architecture for mobile basestation and digital broadcast applications is presented in Figure 1.2. The transmitter consists of two main stages; the exciter and power amplifier (PA) [6, 245]. The exciter stage transforms the binary input data stream (information) into an RF communication signal whilst the power amplifier stage boosts RF signal power to a suitable transmission level.

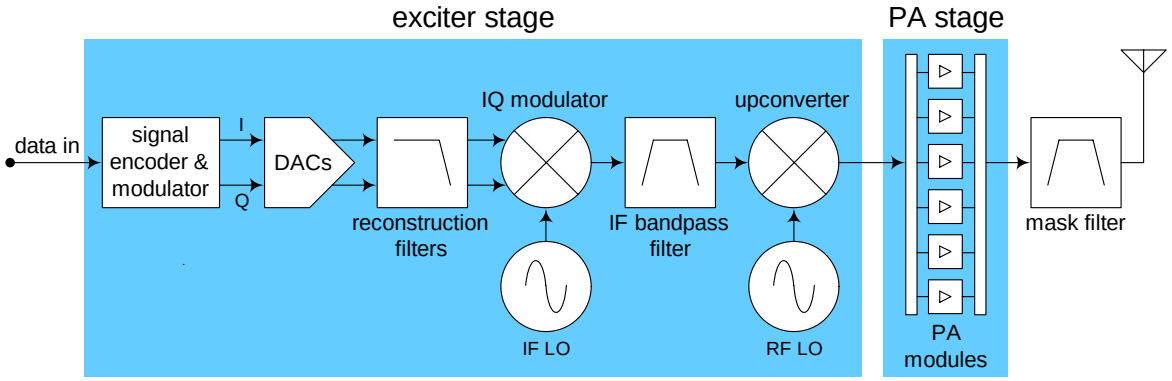


Figure 1.2: Conventional radio transmitter architecture

The exciter stage is implemented as follows. A signal encoder and modulator generates a discrete-time complex baseband communication signal from the binary input data stream. This signal is applied to a pair of DACs and lowpass reconstruction filters to form a continuous-time signal. An IQ modulator then converts this complex baseband signal to a real IF signal which is subsequently translated to RF via another mixing stage. Average power of the exciter output is typically in the order of 10 milliWatts. For 3G basestation applications, the exciter performs DS-CDMA modulation [130, 142] whilst for digital television/radio broadcast applications, it performs OFDM modulation [40, 108].

The power amplifier stage is implemented as a parallel combination of power amplifier plug-in modules. The total number of modules is dependent on the required output power. In a similar fashion to the overall stage, each power amplifier module possesses a parallel subamplifier architecture as shown in Figure 1.3.

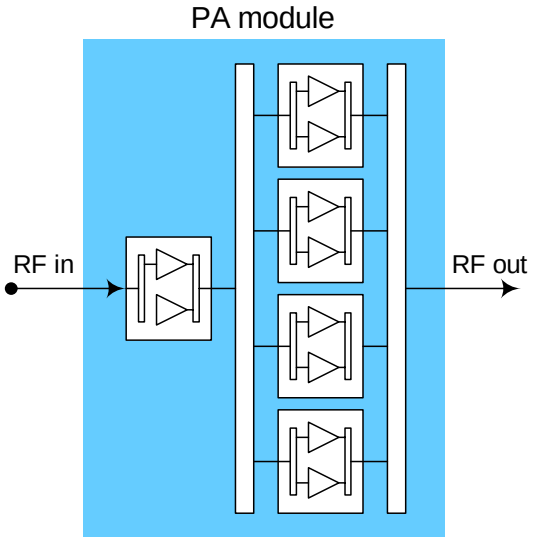


Figure 1.3: Architecture of power amplifier module

This distributed (as opposed to cascaded) approach to amplification allows the use of smaller RF power FETs [60, 201] which offer higher gain, wider bandwidth and better phase linearity [217, 218]. Other advantages of the distributed approach include a soft-failure mode and improved thermal management [237]. For the best tradeoff between efficiency and linearity, each module's final output stage operates in Class-AB, typically push-pull [46, 91]. For mobile basestation applications, average transmitted power is in the order of Watts [1] generally requiring only a single power amplifier module whilst for digital television / radio broadcast applications, it is in the order of KiloWatts [236] requiring several power amplifier modules.

Each power amplifier module, and therefore the entire power amplifier stage, exhibits a nonlinear transfer characteristic due to the RF power FETs and Class-AB output stage. A consequence of this nonlinear transfer characteristic is signal distortion in terms of spectral regrowth [210, 278]. Spectral regrowth is classified as either Co-Channel Distortion (CCD) or Adjacent Channel Distortion (ACD) depending on where it appears in the output spectrum. Figure 1.4 presents the typical output spectrum of a digital television transmitter (DVB-T) prior to mask filtering and without linearization.

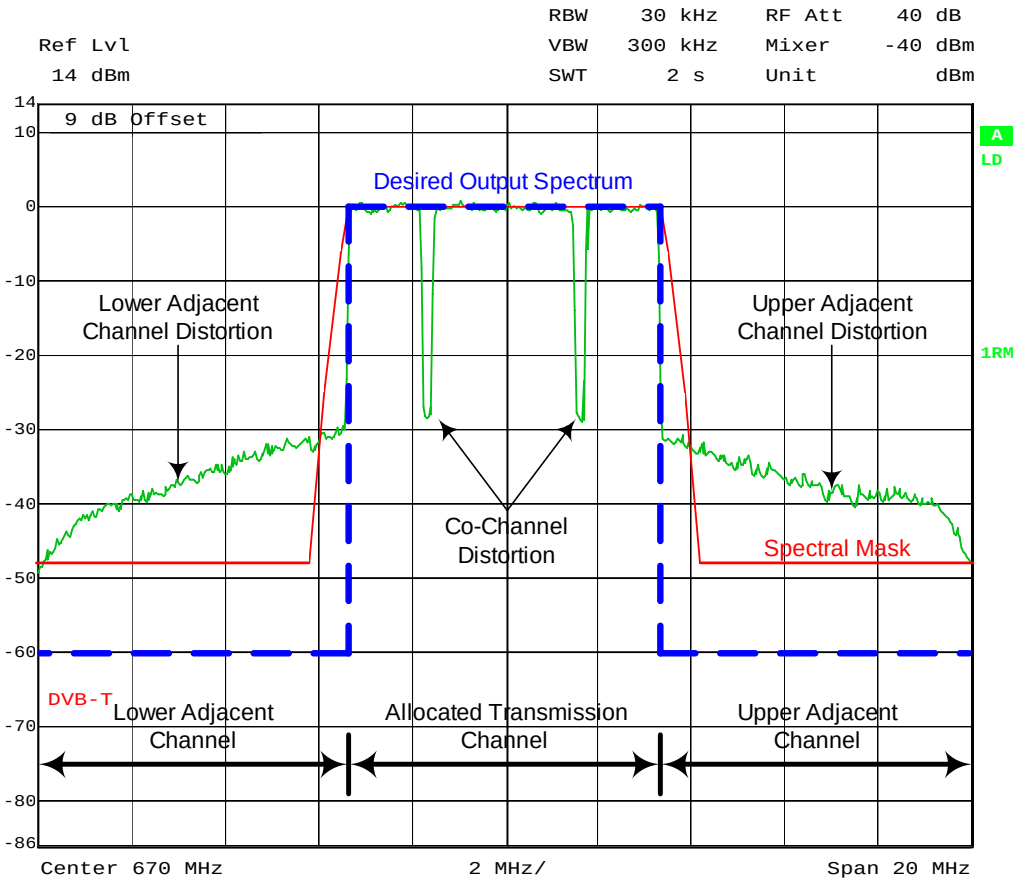


Figure 1.4: Spectral regrowth

The blue dashed trace represents the desired transmitter output spectrum whilst the green trace represents the actual transmitter output spectrum with CCD and ACD marked. Note that in this figure, two sets of OFDM carriers are intentionally removed from the transmission to allow observation of CCD. If these carriers weren't removed, CCD would be obscured by the desired transmission spectrum.

As its name suggests, ACD falls outside the transmitter's allocated transmission channel. The transmitter's bandpass mask filter (refer to Figure 1.2) is tasked with removing this distortion. However, given its finite roll off and attenuation [180,261], some ACD is still transmitted. This distortion ultimately has the potential to interfere with other users of the spectrum. In order to control such interference, government regulatory authorities impose strict spectral emission requirements in the form of spectral masks [2, 64, 65]. The red trace in Figure 1.4 represents the DVB-T mask for the specific scenario of a co-located digital television transmitter. The transmitter's output spectrum must remain within this mask whilst broadcasting otherwise it is deemed noncompliant and penalties apply. ACD is characterized in the frequency-domain by the Adjacent Channel Power Ratio (ACPR) [5].

In contrast to ACD, CCD falls within the transmitter's allocated transmission channel. While a receiver can filter out ACD from the intended transmitter (greater filter selectivity at IF [308]), it cannot filter out CCD. As a result, CCD interferes with the intended transmission, resulting in symbol constellation warping / spreading and therefore symbol detection errors and increased Bit Error Rate (BER) [168]. CCD is characterized by the Error Vector Magnitude (EVM) or Modulation Error Ratio (MER) in the symbol constellation-domain [66]. As alluded to previously, CCD can only be observed in the frequency-domain if a portion of the intended transmission spectrum is removed. For OFDM this means removing carriers whilst for DS-CDMA this means notch filtering. If this is possible, CCD can be characterized in the frequency-domain by the Co-Channel Power Ratio (CCPR) [210].

Modern digital transmission systems utilize OFDM and DS-CDMA signals because of their superior spectral efficiency and multiple access properties respectively [70]. Examples of such systems include digital television (DVB-T), digital radio (DAB) and 3G mobile (UMTS WCDMA). OFDM signals are constructed by adding together many individual carriers (orthogonally frequency spaced) each with a randomly encoded amplitude and phase [100,161] whilst multi-user DS-CDMA signals are constructed by synchronously adding together many chipped RRC pulses, once again, each with a randomly encoded amplitude and phase [77, 271]. This action of adding together many individual signal components each with a randomly encoded amplitude and phase leads to an overall signal with nonconstant envelope and very high Crest Factor, also known as Peak-to-Average Power Ratio (PAPR). Table 1.1 presents a comparison of Crest Factors for various signal modulations.

Signal Modulation	Crest Factor (dB)	Assumptions
FM	0	Theoretical
QPSK	6.2	Random bit stream
16 QAM	8.0	Random bit stream
DAB, DVB-T, WCDMA	≈ 11	Statistically Dependent

Table 1.1: Comparison of Crest Factors for various signal modulations [234]

A clear downside to using these OFDM / CDMA signals is thus the need to accommodate their very high Crest Factors during physical transmission. Compared to other communication signals transmitted with the same average power, these high Crest Factor signals generate greater transmitter distortion (ACD and CCD). This is because their larger signal peaks drive the transmitter power amplifier into regions of greater nonlinearity.

One obvious solution to avoid this increased level of distortion is to back off input signal power to the transmitter power amplifier stage such that each of its modules is always operating in a linear region. This solution, also known as Output Back Off (OBO) [204], is unfavorable however as it significantly reduces the efficiency of each module and necessitates the use of additional parallel modules in order to achieve the required transmission power. This inefficient utilization of the power amplifier ultimately leads to increased operational / acquisition costs, cooling requirements and system footprint. When used, 6–9 dB back off is typical [37].

A more favorable solution, one that avoids OBO, is to incorporate additional signal processing into the transmitter to further linearize the transfer characteristic of the power amplifier stage. Many such linearization techniques exist including Cartesian Loop [211], Envelope Elimination and Restoration (EE&R) [124], feedforward [26,251], negative feedback [27], Linear Amplification Using Nonlinear Components (LINC) [42], RF predistortion [135] and digital predistortion [198, 238]. As a measure of effectiveness of these linearization techniques, WCDMA power amplifiers are estimated to achieve efficiencies of 3-5% with OBO, 6-8% with feedforward linearization, 8-10% with current digital predistortion and greater than 15% (projected) with future digital predistortion [136, 186].

1.2 Digital Predistortion Linearization

For high Crest Factor OFDM and DS-CDMA applications, digital predistortion is both the favored and most promising linearization technique given its cost effectiveness, superior signal processing capability, potential to adapt and ability to linearize the entire transmitter; not just the power amplifier stage. The digital predistortion process involves inserting a nonlinear digital filter directly at the output of the transmitter's signal encoder and modulator as shown in Figure 1.5.

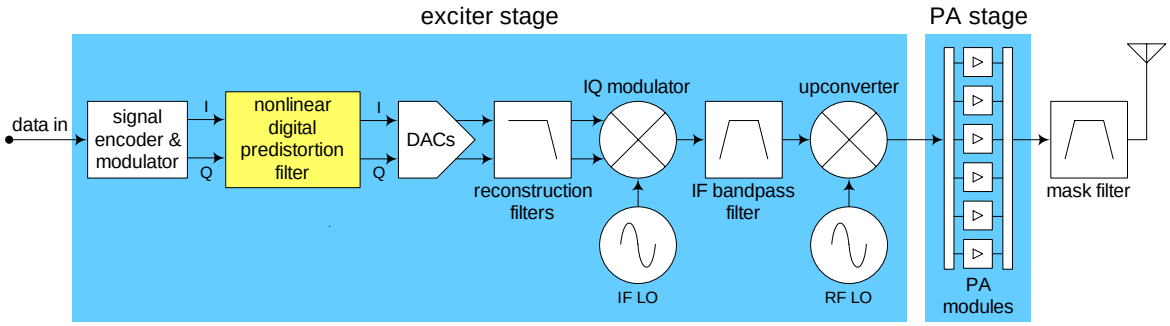


Figure 1.5: Placement of digital predistortion filter

The filter's transfer characteristic is designed to be the inverse nonlinear transfer characteristic of all downstream transmitter components, but specifically the power amplifier stage, thereby creating an overall linear transmission cascade.

Theoretically, digital predistortion can allow a power amplifier to be operated at a higher power with the same level of distortion or at the same power with a lower level of distortion thus allowing a lower order output mask filter to be used to ensure the regulatory mask is met [9]. Generally speaking, the latter scenario is pursued.

1.3 Conclusions Reached

Based on this introductory discussion, the following conclusion is reached:

> There is a greater requirement for transmitter linearization in OFDM / CDMA systems due to the nonconstant envelope and high Crest Factor of such signals, and digital predistortion is the most promising technique to successfully perform this linearization requirement.

Since digital predistortion is the way forward, we next perform a literature review of this technique with the aim of identifying its current state-of-the-art and hence opportunities for further beneficial research.

Chapter 2

Literature Review of Digital Predistortion

Digital predistortion was first reported by *Grabowski and Davis* [89] in 1982. Over the course of three decades, digital communication systems have evolved in terms of signal modulation, signal bandwidth, multiple access strategies and power amplifier hardware [101, 106, 170, 171]. It is this continual evolution of the digital communication system which drives the flow on evolution of digital predistortion and which makes such research still relevant today.

The design and analysis of a digital predistortion system is framed by two fundamental questions:

1. What architecture will the predistortion filter take?
2. How will the characterizing parameters of this filter be estimated?

Before reviewing the literature guided by these two questions, we first introduce the concept of power amplifier memory since this memory has a significant influence on design outcomes.

Concept of Power Amplifier Memory

Ideally, a power amplifier's transfer characteristic is an instantaneous linear gain. In practice however, the power amplifier's output signal is, to some extent, a linear and nonlinear function of both past and present inputs. This undesirable dynamic behavior is referred to as *amplifier memory*. In terms of digital predistortion, it is an amplifier's nonlinear memory components which are of main focus and from this point forward our use of the term *memory* will refer specifically to these nonlinear memory components.

Amplifier memory can be categorized as either *electrical* or *electrothermal*. *Electrical* memory is caused by variation in matching and bias network impedances over

the input signal bandwidth (frequency response), filter group delays, semiconductor charge-carrier trapping effects and transistor nonlinear capacitance while *electrothermal* memory is attributed to transistor junction temperature couplings (dynamic self heating) [28,138,278,279]. *Electrical* memory components can be both short or long-term whilst *electrothermal* memory is generally long-term only [29,48,166,178,228].

The extent of both types of memory is ultimately dependent on the specific design and manufacture of the amplifier as well as modulated signal characteristics such as Probability Density Function (PDF), Complementary Cumulative Distribution Function (CCDF), Crest Factor, bandwidth, average power and carrier frequency [29,105,219,280].

Narrowband Memory

In narrowband applications, where the input signal bandwidth is small compared to the inverse of the amplifier's memory duration, the amplifier is said to be *quasi-memoryless*. This follows from the fact that if we let $x(t)$ be the input temporal signal and τ be the amplifier memory duration, then $x(t-\tau) \approx Kx(t)$ where K is a complex coefficient and hence all memory terms within the amplifier's nonlinear input-output functional can be packaged into an equivalent accumulated instantaneous effect.

Quasi-memoryless amplifiers are characterized by the Amplitude Modulation – Amplitude Modulation (AM-AM) and Amplitude Modulation – Phase Modulation (AM-PM) responses. The AM-AM and AM-PM responses of an amplifier are by definition the *output signal amplitude* and *output signal phase* measured as a function of *input signal amplitude*, in both cases with sinusoidal excitation [146,222]. These responses capture the amplitude and phase distortion of the amplifier, resulting from nonlinear narrowband memory.

Wideband Memory

In wideband applications, where the temporal variation of the input signal envelope occurs well within the amplifier's memory duration, the previous approximation does not hold and the amplifier must be considered purely dynamic, in which case AM-AM and AM-PM characterizations of the amplifier are no longer appropriate. In fact, scattering (hysteresis) will occur in any AM-AM and AM-PM response as demonstrated in [80,139].

In these wideband scenarios, power amplifier memory manifests itself as frequency dependent behavior, specifically amplitude and phase asymmetries between the lower and upper spectral regrowth sidebands. This asymmetry is dependent on both carrier frequency and envelope bandwidth [29,147,155].

The simplest laboratory based method for characterizing these asymmetric memory effects is to use a two-tone signal. The two-tones, with varied tone spacing, are applied to the amplifier and the distortion components are measured. If the distortion components are identical in amplitude and phase regardless of the tone spacings, then the amplifier is considered memoryless otherwise it is considered dynamic [34,146,166]. This process is demonstrated in [34,139,188]. Two-tone testing is merely a simplification test case of the process of applying a multicarrier or wideband digitally modulated signal to the amplifier. In either case, higher spurious emission generally occurs in the lower frequency sideband [139].

Wideband memory effects are also known as *dynamic distortions* within the literature and are one of the limiting factors for distortion cancellation in power amplifier predistortion [80].

Building on this introduction to power amplifier memory, we now proceed to review the literature guided by the two fundamental questions on Page 9. As will become evident, the outcome of both questions is strongly influenced by the communication system bandwidth in question. That is, the overall predistorter design is different for narrowband compared to wideband applications.

2.1 Predistortion Filter Architectures

Early Narrowband Applications

The majority of early work in digital predistortion is associated with *microwave* high power terrestrial link and satellite / earth station transmitters generally composed of TWT or GaAs FET amplifiers with high order linear signal modulation, typically 16 - 256 QAM [71,90,127,131,216,254]. Compared to analog FM or digital PSK / MSK / CPM, this signal modulation has higher spectral efficiency but a non-constant envelope which inevitably leads to nonlinear amplifier distortion. Given the narrow percentage bandwidth of these high frequency systems, transmitters are considered *quasi-memoryless* and as a consequence the complementary predistorter nonlinearity is also treated as a memoryless system [216,222]. These instantaneous predistorters fall into two general categories, *data* and *signal* predistorters, depending on whether the predistorter is placed before or after the pulse shaping filter respectively [49].

2.1.1 Data Predistortion (Narrowband)

Data predistortion [215,242], also known as *prerotation*, involves modifying the shape of the input symbol constellation such that after passing through the nonlinear power amplifier stage, the output constellation is the desired original shape. It naturally follows that *data* predistortion is highly restricted by pulse shaping format and has constant bandwidth occupancy. If *raised cosine* (Nyquist) pulse shaping is utilized, *data* predistortion has the ability to correct constellation warping but has no effect on spectral regrowth. In contrast, if *root raised cosine* pulse shaping is utilized in order to establish matched filtering across the communications channel, *data* predistortion is unsuitable due to the pulses being nonzero at adjacent symbol periods. For the same reason, memoryless transmitter nonlinearity coupled with *root raised cosine* pulse shaping leads to intersymbol interference and hence constellation spreading. More advanced versions of the *data* predistortion technique strive for better performance by incorporating memory [129, 150, 214] or interpolation [128]. In these respective techniques, constellation predistortion is dependent on preceding / succeeding symbols and is performed several times within a symbol interval.

2.1.2 Signal Predistortion (Narrowband)

Signal predistortion acts on the full input signal modulation, not just its constellation, as is the case with *data* predistortion. *Signal* predistorters can be further categorized as either *mapping* or *complex gain* predistorters [67, 69, 174]:

The *mapping* predistorter [182, 189] indexes its IQ input and maps each such index to a corresponding IQ output. The two dimensional mapping space is derived via comparison of the baseband input and demodulated feedback and ultimately stored in a two dimensional (IQ) Random Access Memory (RAM) Look Up Table (LUT). In effect, this predistorter directly maps the baseband complex plane to the RF complex plane and is therefore capable of correcting both the envelope distortion of the amplifier (AM-AM / AM-PM) as well as the cartesian nonidealities associated with the modulation process which include carrier leakage, mixer distortion and mismatches in the analog IQ channels [33, 68, 69, 270]. The disadvantages of this approach are the excessive number of indices or subsequent trade-off interpolation required as well as slow convergence rate. [32] provides an analysis of the errors associated with this input quantization.

The *complex gain* predistorter [32, 86, 290] is concerned solely with linearizing the power amplifier, not the entire transmitter, as is the case with *mapping* predis-

tortion. As a consequence, this predistorter only indexes its input signal complex envelope (one dimension rather than two) and applies a complex gain correction based on the amplifier's AM-AM / AM-PM characteristic. This in effect gives the transmitter a constant power gain with respect to input signal power. Since indexing is one dimensional, the size of the associated RAM LUT is considerably smaller and interpolation is simplified [269] however conversion to / from rectangular / polar coordinates as well as complex gain implementation requires additional processing. As one would expect, since cartesian modulated errors are not corrected for, poorer performance generally results for *complex gain* predistorters [174]. Adaptive versions of this technique are also proposed [119, 152] along with statistically weighted variants [212]. Direct computation of the complex gain correction via cubic spline interpolation has also been proposed as an alternative to LUT indexing [3].

Analog Predistortion (Narrowband)

It is worth noting that *analog IF* predistortion [110, 191, 192] and *analog RF* predistortion [47, 137, 273] are also proposed for early narrowband applications. However in addition to being more complex, expensive and dependent on final frequency, the performance of these predistorters is generally poorer than their digital counterparts due to their hardware inflexibility and restrictive 3^{rd} order transfer characteristics based on active devices or Schottky and Varactor diodes [25, 208]. For FET based power amplifiers, Class AB configurations have significant kinks in their transfer characteristic which are not well modeled by a cube law [31]. Similarly, 3^{rd} order distortion components generated by individual stage transistors combine and intermodulate with each other at the output stage to create higher order components, hence requiring higher order predistorters [201].

Later Wideband Applications

The majority of later digital predistortion research is associated with digital cellular and broadcast transmitters. This is a direct consequence of the enormous growth and evolution of digital communication systems over the last decade, particularly into the consumer marketplace [250, 259]. LDMOS FET amplifiers are of prime focus along with WCDMA and OFDM signal modulation [126, 212], due to the widespread and current use of the UMTS and DVB / DAB standards. Compared to narrowband microwave, these more recent applications have significantly lower carrier frequencies (L-Band and UHF respectively) and therefore larger percentage bandwidths, making them wideband in nature. Memory effects are therefore exhibited by transmitter power amplifiers (as discussed previously) and hence more

complex predistorters possessing memory are now required as the memoryless compensation techniques already developed prove ineffective [28, 99, 138]. [85] demonstrates with spectral screenshots the effect that insufficient predistorter memory has on the transmitter output spectrum, namely reduced inband linearization performance and asymmetric lower / upper adjacent channel power. Compared to pure linear signal modulation utilized by earlier microwave transmission applications, these new modulation formats exhibit higher signal Crest Factors and hence pose an even greater requirement for digital predistortion to balance the power amplifier efficiency / linearity trade off [29, 183, 226].

Predistortion filter architectures proposed for wideband applications possess some form of memory and are behavioral data filters [113] rather than physical or equivalent circuit representations [293]. This trend towards behavioral modeling is driven by the lower complexity and processing requirements hence higher achievable data throughput and ease of repetitive simulations [117,209,257]. The accepted choice of behavioral architecture is one that, at minimum, accurately characterizes the power amplifier and hence has the capacity and potential to generate an equivalent canceling characteristic [203]. Proposed behavioral architectures can generally be categorized as either *advanced* or *conventional* [153]:

Advanced architectures are based on neural networks [24, 162, 282, 294] or fuzzy methods [153, 158]. Due to their poor analytic nature and slow convergence, these architectures receive limited coverage in the literature and are nonexistent in commercial implementation.

Conventional architectures are based on nonlinear polynomials or power series [121, 144]. They include the Volterra Series, Memory Polynomial, Nonlinear Auto-Regressive Moving Average (NARMA) filter, Hammerstein / Wiener filters, Twin Nonlinear Two-Box (TNTB) model, as well as variants and hybrids of all [80, 274, 289]. These architectures dominate the literature and are discussed below.

2.1.3 Volterra Series (Wideband)

The discrete-time, causal, complex baseband Volterra Series model with maximum nonlinearity P (odd) and memory M is given by:

$$y[n] = x[n] + \sum_{a=1}^{\binom{P-1}{2}} \left[\sum_{k_1=0}^{M} \cdots \sum_{k_{2a+1}=0}^{M} \left(h_{2a+1}[k_1, \ldots, k_{2a+1}] \prod_{i=1}^{a+1} x[n-k_i] \prod_{j=a+2}^{2a+1} x^*[n-k_j] \right) \right] \tag{2.1}$$

In this equation, $x[n]$ and $y[n]$ represent the input and output complex signal envelopes respectively and $x^*[\cdot]$ denotes complex conjugation. $h_{2a+1}[k_1, \ldots, k_{2a+1}]$ is

called the $(2a+1)^{\text{th}}$ order Volterra kernel and the entire set of kernels fully characterizes the model [248]. It is noted that this baseband model only contains odd order kernels. This is because spectral regrowth created by even order kernels is far displaced from the allocated transmission channel at RF and subsequently removed by mask filtering at the output of the transmitter. It is also noted that the Volterra kernels are complex, containing real and imaginary parts. Being a sum of multidimensional convolutions, this baseband Volterra model is the most general and comprehensive behavioral model for dynamic predistortion systems with all other behavioral models, including neural networks and fuzzy methods, being specific cases [289]. The drawback to such a powerful model however is its kernel size. The number of kernel coefficients which need to be estimated increases exponentially with the degree of nonlinearity and memory length [202, 300, 301, 305]. To overcome this drawback, the baseband Volterra model is judiciously pruned to retain only those kernel coefficients that are expected to influence performance [301]. Numerous pruning strategies are proposed including *Dynamic Deviation Reduction* [302–304], *Physical Knowledge* [305], *V-Vector Algebra* [30,298,306], *Near-Diagonality Restriction* [299], *Base-Band Derived Volterra* [160], *Dynamic Volterra* [151,199] and *Volterra Behavioral Wideband* [45]. This reduction in complexity increases the Volterra model's practical use from weak to mild and strongly nonlinear systems. An alternative technique for reducing kernel size, one that does not involve pruning, has also been proposed. This involves projecting the Volterra Series onto a set of Laguerre or Kautz orthonormal basis functions, making the number of estimated parameters independent of memory length [97, 114, 115, 300]. Spline basis functions have also been proposed as a more efficient approximation [241] though basis function projections have seen limited uptake.

2.1.4 Memory Polynomial (Wideband)

The complex baseband Memory Polynomial [22, 57, 58, 138], also known as the Nonlinear Moving Average (NMA) filter [117, 184], is given by:

$$y[n] = x[n] + \sum_{a=1}^{\binom{P-1}{2}} \left[\sum_{k=0}^{M} h_{2a+1}[k] \left| x[n-k] \right|^{2a} x[n-k] \right] \tag{2.2}$$

With the same notation as (2.1), it is the simplest of the conventional power series models and corresponds to the Volterra Series with diagonal terms only. In this sense it is yet another pruned Volterra model but common enough to stand on its own. An orthogonal variant known as the *Generalized* Memory Polynomial also exists [186]. Whilst both forms are theoretically equivalent, the difference lies in higher order numerical stability (matrix inversion operations) with the orthogonal

variant performing better in the presence of quantization noise and finite precision processing [220, 221]. The *Envelope* Memory Polynomial, another common augmentation of the pure Memory Polynomial, simplifies implementation by replacing IQ memory terms with magnitude only memory terms. With this trade off towards simplicity, nested LUT implementation becomes possible [55, 99].

2.1.5 NARMA Filter (Wideband)

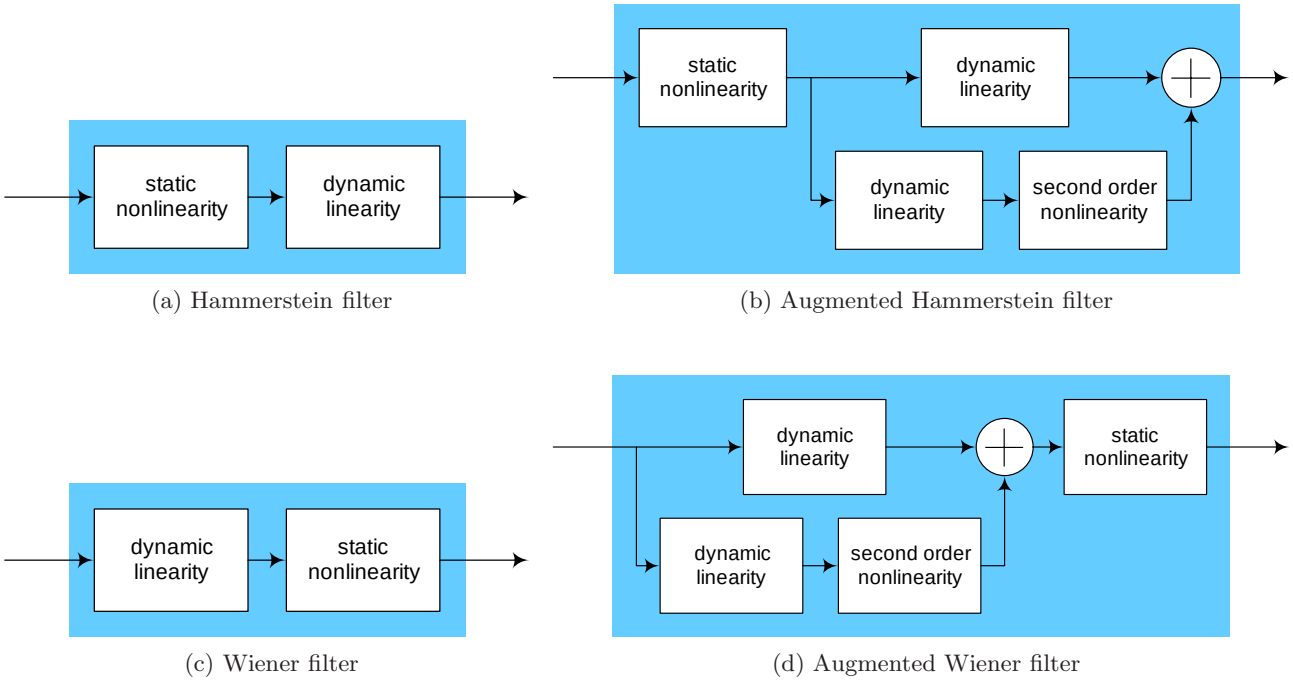
The Nonlinear Auto-Regressive Moving Average (NARMA) filter extends the concept of the Memory Polynomial by incorporating diagonalized output memory terms within the filter's functional power series [85, 184]:

$$y[n] = x[n] + \sum_{a=1}^{\binom{P-1}{2}} \left[\sum_{k=0}^{M} h_{2a+1}[k] \left|x[n-k]\right|^{2a} x[n-k] + \sum_{j=1}^{M} b_{2a+1}[j] \left|y[n-j]\right|^{2a} y[n-j] \right] \tag{2.3}$$

The relationship between the Memory Polynomial and NARMA model of the nonlinear world is hence akin to the relationship between the Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters of the linear world [183]. The NARMA structure can reduce the number of coefficients needed to represent short and long term memory effects, though it can also become unstable due to the nonlinear IIR terms [82].

2.1.6 Hammerstein and Wiener Filters (Wideband)

Whilst the previous three architectures attempt to package both dynamic and nonlinear behaviors into a single power series, the Hammerstein and Wiener filters separate these behaviors into two standalone subfilters; a static nonlinearity and a dynamic linearity. As presented in Figures 2.1a and 2.1c, the Hammerstein filter consists of a cascaded static nonlinearity and dynamic linearity whilst the Wiener filter consists of the same cascade appearing in reverse order [81, 92, 93]. This decoupling of nonlinear and dynamic behavior simplifies the parameter estimation process. The static nonlinear subfilter can be implemented as an AM-AM / AM-PM based LUT or instantaneous nonlinear power series whilst the dynamic linear subfilter can be implemented as an FIR or IIR LTI filter [83]. Since the Hammerstein and Wiener filters are structural inverses of each other, they generally appear as matching pairs in predistortion linearization applications. That is, the predistorter will be modeled as one form with the power amplifier modeled as the other. Both H-W and W-H sequences have been proposed. The H-W sequence is applicable to TWT transmitters only, where the combined amplifier and preceding pulse shaping / channel filters are suitably modeled as a Wiener system. The advantage of this sequence is that it is possible for the Hammerstein predistorter to be an exact inverse of the power



(a) Hammerstein filter

(b) Augmented Hammerstein filter

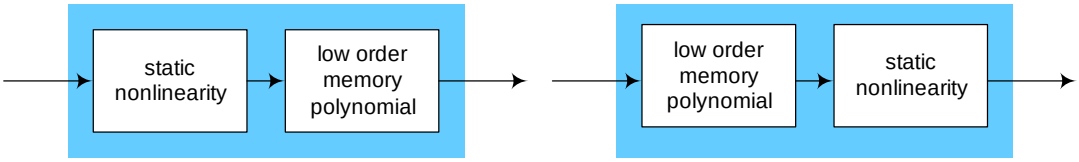
(c) Wiener filter

(d) Augmented Wiener filter

Figure 2.1: Hammerstein, Wiener and Augmented filter architectures

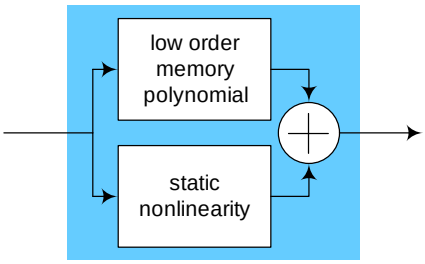
amplifier given its pre-equalizing connection of LTI subfilters [35, 56, 58, 126]. For all other transmitter cases, the W-H sequence tends to be applied. Despite an exact inverse relationship not existing and parameter estimation reported to be more difficult, the W-H sequence tends to apply to broader transmitter classes and hence can compensate amplifier memory effects more effectively all round [281, 292].

Extensions of the conventional Hammerstein and Wiener predistorter architectures have also been proposed. These extensions attempt to address additional bias / harmonic loading electrical memory effects and hence increase performance. As presented in Figures 2.1b and 2.1d, the Augmented Hammerstein [164, 165] and Augmented Wiener filters [163] add an additional parallel branch to their dynamic linear filters. This branch consists of another dynamic linearity cascaded with a second order nonlinearity hence transforming the original subfilter into a mildly dynamic nonlinearity with even order components. It is claimed that the subsequent even order distortion remixes with the carrier frequency, leading to odd order canceling products within the main channel [163]. Other variants of the conventional Hammerstein and Wiener architectures are also proposed in the literature though all instances are applied only to power amplifier modeling, not predistortion. For reference, these architectures include the Extended Hammerstein [116], Parallel Wiener [145,258], Combined Wiener-Hammerstein [43,186,243] and Parallel-Cascade filters [155].



(a) Forward TNTB model

(b) Reverse TNTB model



(c) Parallel TNTB model

Figure 2.2: Twin Nonlinear Two-Box (TNTB) model architectures

2.1.7 TNTB Model (Wideband)

The final wideband predistortion filter architecture is the Twin Nonlinear Two-Box (TNTB) model. Similar to the conventional Hammerstein / Wiener filters, the architecture consists of two subfilters. The difference however lies in replacing the conventional dynamic linearity with a low order nonlinear Memory Polynomial, in effect extending the idea of the Augmented Hammerstein / Wiener variants [80]. Three classes of TNTB models exist as presented in Figures 2.2a, 2.2b and 2.2c. In the forward, reverse and parallel models, the static nonlinearity is placed upstream, downstream and in parallel to the Memory Polynomial respectively. This model is reported to out-perform the conventional stand-alone Memory Polynomial architecture whilst reducing the number of model parameters by up to 50% [98].

The previous sections have reviewed both narrowband and wideband predistortion filter architectures (Question 1, Page 9). Our review now turns attention to predistortion filter parameter estimation techniques (Question 2, Page 9).

2.2 Predistortion Filter Parameter Estimation

Two general strategies exist for estimating predistortion filter parameters; these being *Model Based Derivation (MBD)* and *Self Learning*. The two strategies differ fundamentally in how they *acquire* and *utilize* knowledge of the power amplifier's nonlinearity. That is, the *MBD* strategy relies on power amplifier modeling and subsequent mathematical inversion to form its predistorter estimate whereas the

Self Learning strategy relies on power amplifier feedback and subsequent predistorter self tuning. Depending on the form of the feedback, the *Self Learning* strategy can be further refined as either *Indirect Learning*, *Direct Learning* and *Spectral Power Feedback Learning*. These strategies are discussed below.

2.2.1 Model Based Derivation (Narrowband / Wideband)

In its most general form, an *MBD* strategy is a two step process as presented in Figure 2.3. The first step consists of modeling the power amplifier using system identification techniques. The second step consists of deriving the predistortion filter from the power amplifier model.

For memoryless power amplifiers, both steps are routine with all of the narrowband microwave techniques discussed earlier (*Data*, *Mapping* and *Complex Gain*) being successful examples of this strategy.

The same cannot be said however for wideband applications where power amplifiers exhibit memory effects. In addition to being a more complex system identification problem, derivation of the predistortion filter from the power amplifier model is significantly more involved [138, 303]. For this reason, the *MBD* strategy is not generally as common in wideband applications where memory effects must be taken into account.

It is noted that *MBD* has also been referred to as *Direct Learning* in specific parts of the literature [206]. This creates potential confusion since one of the prominent *Self Learning* strategies (to be discussed soon) is also called *Direct Learning*. To be consistent with the majority of literature, we do not mix the terms *MBD* and *Direct Learning*.

The first step of *MBD* (modeling the power amplifier using system identification techniques) involves choosing a nonlinear behavioral model to adequately represent

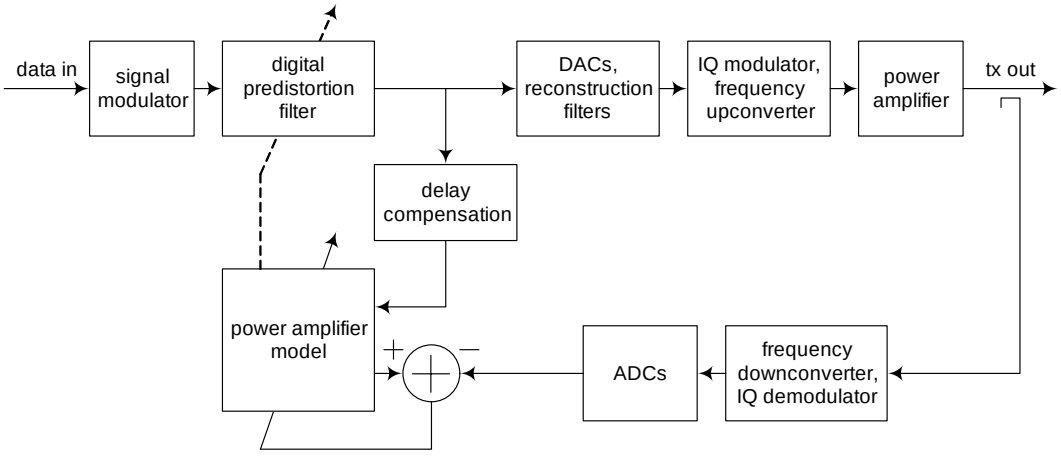


Figure 2.3: Model Based Derivation (MBD) strategy

the power amplifier, relinearizing this model's input-output equation and then estimating the resulting linear parameters via vectorized linear regression techniques based on power amplifier input and output signals [160]. Such techniques include Least Mean Squares (LMS), Recursive Least Squares (RLS) and Fast-Kalman filtering [8,23,38,94,169,173,176]. Relinearization specifically involves creating vectorally separate signal streams composed of the original signal raised to the appropriate power. It must be understood that this can only be performed if filtering is linear with respect to kernel elements, despite the filter output being nonlinearly combined with the input. This is true for all behavioral models presented previously and is hence assumed in the following review.

This first step of *MBD* was pioneered by the power amplifier modeling community for communication system simulation but later recognized as being applicable to the digital predistortion community's *MBD* strategy. It is worth noting that *Polyspectral* techniques have also been proposed for modeling dynamic nonlinear systems [255–257] however specific application to *MBD* predistorter estimation is still in its inception.

The second step of *MBD* (deriving the predistortion filter from the power amplifier model) is essentially a mathematical inversion problem. Several techniques have been proposed for this, including P^{th} *Order Inverses*, *Fixed-Point Iteration* and *Model Rearrangement*.

The *P^{th} Order Inverses* technique is by far the most common [25, 190, 297, 303]. Pioneered by *Schetzen* [247, 248], the technique requires the power amplifier to be modeled as a Volterra system in the first step of *MBD*. By definition, the P^{th} order inverse predistorter is then analytically computed as another Volterra system which removes nonlinear transmitter distortion up to the P^{th} order. Theoretically, nonlinear distortion greater than P^{th} order remains which must be factored into the design process. Despite being mathematically rigorous, a major disadvantage of this technique is the computational complexity involved in both the Volterra system identification process and also the analytical inverse computation [272], leading to non-real-time operation. To alleviate this and achieve real-time performance, the Volterra model must be aggressively pruned in kernel degrees-of-freedom, memory and nonlinear order which inevitably leads to reduced performance.

The *Fixed-Point Iteration* technique requires the power amplifier to be modeled as a nonlinear AM-AM / AM-PM characteristic in the first step of *MBD*. The mathematical inverse of this model is then computed via standard numerical *Fixed-Point Iteration* [132,223]. Mathematically speaking, the uniqueness of the numerical solution is validated by the *Contraction Mapping Theorem* [167,207]. Despite being

proposed for OFDM and CDMA applications [112, 140, 291], the wideband performance of this technique is limited by the memory constraints of its power amplifier model.

The *Model Rearrangement* technique requires the power amplifier to be modeled as a simple dynamic polynomial whose form specifically allows model input to be rearranged in terms of past model inputs and model output. Following substitution of the model output with the conditioned transmitter input (this being the ideal transmitter output after linearization), an expression for the dynamic predistorter input-output equation remains since the model input is merely the predistorter output. This expression can then be solved via vector linear regression techniques [85, 183, 184] or iteration [138]. Despite adaption being possible, the downfall of *Model Rearrangement* is the primitive dynamic polynomial model required to allow rearrangement to occur. This leads to questionable model fidelity and is a perfect example of how the *MBD* strategy is not well suited to wideband applications.

2.2.2 Indirect Learning (Wideband)

The *Indirect Learning* strategy was first proposed by *Eun and Powers* [63]. Its architecture is presented in Figure 2.4. The term *indirect* follows from the predistortion filter being derived *indirectly* from a postdistortion filter. Referring to Figure 2.4, the power amplifier output is attenuated, downconverted / demodulated and fed to a nonlinear filter (postdistortion filter). This filter's architecture is identical to the proposed predistortion filter. Following a vectorized mathematical relinearization of the postdistortion filter's input-output relationship, its parameters are estimated via linear regression to minimize the transmitter's output distortion. Once converged, the postdistortion filter's parameters are copied to the predistortion filter [62, 79, 156, 157]. Based on its architecture, this strategy is also known as

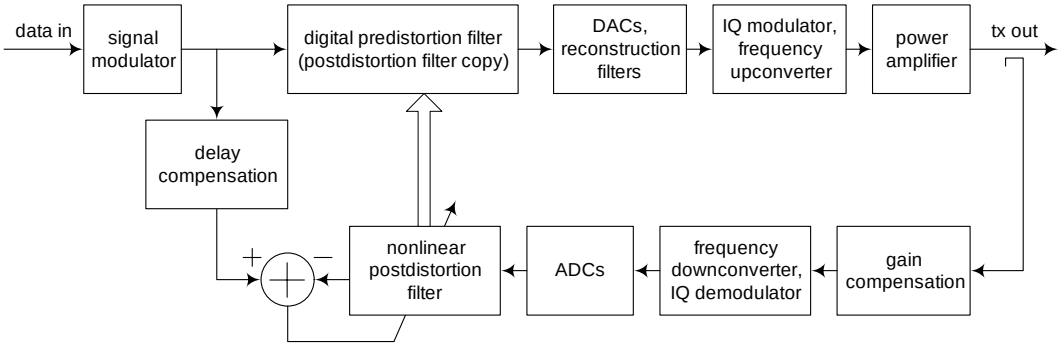


Figure 2.4: Indirect Learning strategy

Postdistortion and Translation in the literature.

Whilst intuitive and appealing, the problem with this strategy is its assumption of nonlinear system commutation, that is, the equivalence of postdistortion and predistortion. Whilst the commutative property may hold for cascaded linear systems, it doesn't generally hold for cascaded nonlinear systems and hence the translation from postdistortion to predistortion can only be considered an approximation without guarantee of adequate performance [81, 184, 206, 295]. [186] points out specific cases to the contrary [248] however these cannot be relied upon in the general sense.

Another problem with the standard *Indirect Learning* strategy is that it's not capable of seamless adaptive predistortion. That is, predistortion filter adaptation can only be performed by turning off the current predistortion filter so as to allow fresh transmitter output data to be collected for postdistortion linear regression. This in turn means repetitive periods of unlinearized transmission (high distortion) whilst the predistortion filter is turned off [80]. To overcome this downfall, [59, 187] incorporate a power amplifier model. This model is continuously updated via a linear system identification technique and allows the postdistorter to be estimated and translated continuously thereby achieving seamless adaption. The questionable aspect of this however is the accuracy of the power amplifier model and its flow on effect in estimating the postdistortion filter. Another method to achieve adaption is proposed by [177] whereby the output of the predistortion filter (as opposed to input) is used in the optimal tuning of the postdistortion filter. This allows the postdistortion filter to be estimated and continuously translated to the predistortion filter whilst on-air. The downside to this approach however is the increased complexity in loop delay compensation (frequency dependent) given the greater signal bandwidth resulting from predistortion spectral regrowth.

2.2.3 Direct Learning (Wideband)

As its name implies, the *Direct Learning* strategy is a more direct approach to *Self Learning*. Used by transmitter manufacturers such as Rohde & Schwarz, NEC, Ericsson and Harris [102,103,194,195,236,237], its architecture is presented in Figure 2.5. Following a relinearization of the predistortion filter's input-output equation, vectorized linear regression parameter estimation is performed with the objective being to minimize the mean square error between the baseband attenuated transmitter output and delayed input [22]. This minimization is equivalent to removing transmitter output distortion.

Compared to the *Indirect Learning* architecture, the nonlinear power amplifier is now incorporated into the linear regression optimization process. While this leads to the distinct advantages of being more direct and avoiding the nonrigorous commutation assumption, it leads to three major problems of its own:

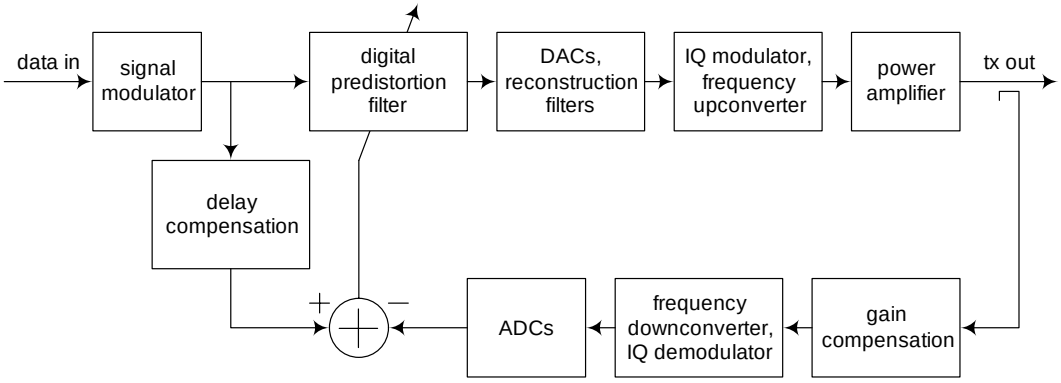


Figure 2.5: Direct Learning strategy

1. Computation and implementation cost is increased significantly [159]. Linear regression gradient estimation must be performed either analytically, whereby a nonlinear model of the power amplifier must first be derived and generally adapted similar to the first step of *MBD* [125, 126, 295], or non-analytically whereby predistorter parameters are incremented in a *finite-differences* sense to probe the power amplifier's response. This probing must be controlled to prevent unintentional on-air distortion in the adaption process. In the *Indirect Learning* strategy, analytical gradient estimation is routinely performed based on the known postdistortion filter architecture.

2. The inherent assumption of a quadratic error criterion does not hold in the local linear regression optimization process [22, 78, 177]. In linear regression theory, the error criterion can only be assumed quadratic if the signal being subtracted from the desired response (to form the error signal) is the direct (or linearly filtered) output of the filter being regressed [104, 285]. In the *Indirect Learning* and *MBD* strategies, this is indeed true and hence they can assume a quadratic criterion. In the *Direct Learning* strategy however, the nonlinear amplifier sits between the signal being subtracted from the desired response and the filter being regressed (predistorter). As a result, the quadratic assumption does not hold and the proposed local optimization process cannot guarantee convergence to the global minimum, with concerns remaining about local inferior convergence.

3. Delay and gain compensation must be implemented within the feedback paths forming the error criterion as shown in Figure 2.5. While not such a problem for narrowband systems, this poses a significant problem for wideband systems where compensation becomes frequency dependent. This point is reinforced by the fact that signal bandwidth in the output feedback path is at least nine times that of the original wideband modulated signal given its adjacent

channel distortion components. If compensation is not performed correctly, the error criterion on which to adapt does not represent the true state of linearization and the optimization process converges to a biased and hence performance degrading value [84, 119, 288, 290]. Time delay mismatch from the input feedback path can also be misinterpreted and considered as memory effects [80, 172]. Typically, the resolution for delay alignment is lower than the signal sampling rate. This requires signal up-sampling and down-sampling during the delay estimation and compensation process [163]. [288] also points out that with the conventional feedback path, changes in temperature for example may cause the power amplifier gain to change which then makes the feedback power level adjustment suboptimal. This means that not only must one adapt for changes in the power amplifier, one must also follow up by adapting components in the feedback path, all of which are dependent on each other, leading to stability issues. While these feedback loop impairments must be equally managed in the *Indirect Learning* strategy, errors associated with the postdistortion translation process appear to attract greater scrutiny. As an aside, frequency dependent modulator / demodulator and DAC / ADC errors also degrade predistortion performance where feedback is concerned. While being more of an early design aspect rather than a learning strategy issue, measures must be taken to avoid excessive performance degradation [33, 68, 226, 253, 268, 270].

Combined Indirect & Direct Learning

Combining both the *Indirect* and *Direct Learning* strategies has also been proposed in [35, 36]. Here, the predistorter is first estimated *Indirectly* and then refined *Directly*. Despite exhibiting improved convergence properties, this combined learning strategy has seen limited uptake most likely due to its doubled implementational cost.

2.2.4 Spectral Power Feedback Learning (Wideband)

To specifically overcome the third pitfall of the *Direct Learning* strategy, that is time-domain feedback loop compensation error, *Stapleton et al.* [107, 262–264] propose working in the frequency-domain with their *Spectral Power Feedback Learning (SPFL)* strategy. Demonstrated in Figure 2.6, the power within a small spectral region of the transmitter's adjacent channel is measured and used as the *objective function* for a *generic single objective optimization* of the predistortion filter parameters [75,200]. Since time delay and power level compensation is avoided, a far more robust, reliable and simple measure of transmitter output nonlinearity results.

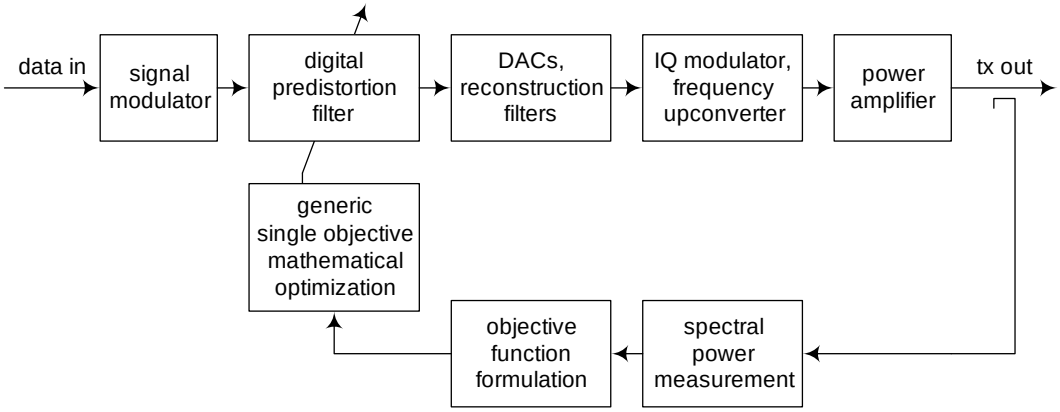


Figure 2.6: Spectral Power Feedback Learning (SPFL) strategy

The key point to understand here is that the *Direct Learning* strategy implements time-domain *signal feedback* whereas this strategy implements frequency-domain *information feedback*; the difference being the elimination of time delay and power level compensation (and associated error) in the feedback path. Spectral power is derived by bandpass filtering and power detection although spectral convolution has also been proposed [265].

The *SPFL* strategy should not be confused with implementations which use ACP measurements for distortion monitoring and adaption control purposes but then revert back to the time-domain *Direct Learning* strategy to perform actual parameter estimation [112, 120, 194]. *SPFL* uses spectral power measurement as the objective in the parameter estimation process.

Another advantage of *SPFL*, being based on generic nonlinear optimization, is that it inherently assumes a nonconvex objective function and hence can potentially use both global and local optimizers to achieve convergence. This eliminates the second pitfall of the *Direct Learning* strategy.

2.3 Opportunity For Further Beneficial Research

Figure 2.7 summarizes the findings of this literature review in terms of system bandwidth, predistortion filter architecture and predistortion filter parameter estimation strategy. Section references are also included to aid navigation within the review.

From a research perspective, the aim of this review is to identify the current state-of-the-art of digital predistortion and hence identify opportunities for further beneficial research. We identify the *SPFL* strategy as the basis of such beneficial research for the following reasons:

1. Since the strategy implements frequency-domain *information feedback*, and is hence not limited by compensation error associated with time-domain *signal*

feedback, it appears more suited to current and future wideband applications than does the *Direct* and *Indirect Learning* strategies. It can only be expected that the continual increase in signal bandwidth will further expose the feedback weaknesses of the *Direct* and *Indirect Learning* strategies.

2. *Stapleton et al.* [107,262–264] have only presented the strategy in terms of systems with linear signal modulation (QAM / QPSK). To the best of our knowledge, no further investigations have been performed for wider-band, higher Crest Factor applications such as CDMA and OFDM.

Based on the above reasoning, we specifically endeavour to investigate the *SPFL* strategy's promising potential to linearize current and future wideband communication systems.

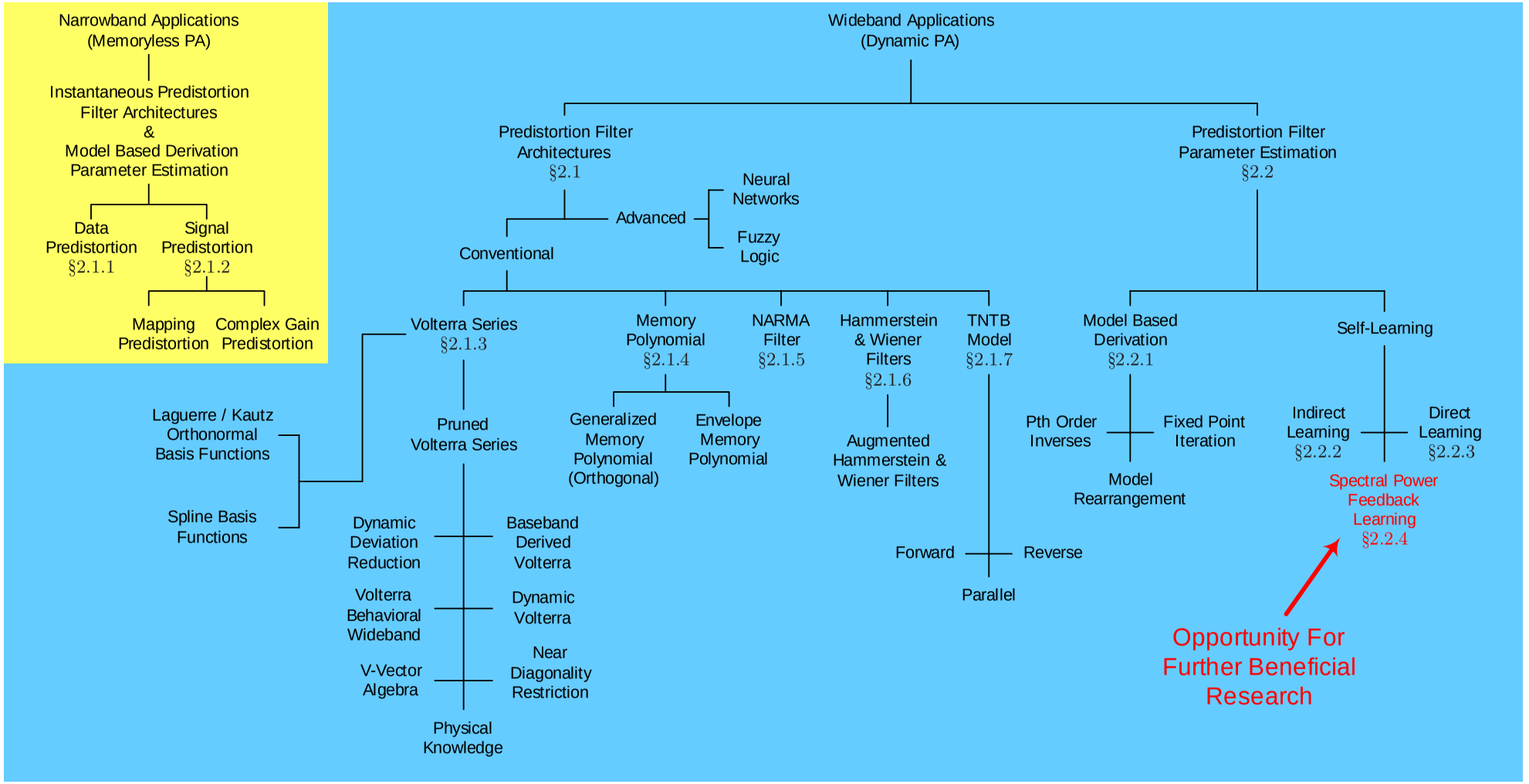


Figure 2.7: Summary of literature review

Chapter 3

Research Scope & Outputs

In this chapter, the scope of the research endeavor will be discussed. In this context, the scope will include the statement of research, a list of target applications, all assumptions and finally the delimitations [154]. After reading this chapter, one will have a clearer understanding of the boundaries of this research.

3.1 Statement of Research

This research will demonstrate adaptive digital predistortion for wideband high Crest Factor applications based on the specific concept of *Spectral Power Feedback Learning*. This encompasses both aspects of predistortion filter *architecture* and *parameter estimation*.

3.2 Target Applications

In terms of the above research statement, *wideband high Crest Factor applications* specifically refers to the following medium to high power (Watt – KiloWatt) radio transmitter standards:

- 3G UMTS WCDMA mobile basestation downlink
  - Modulation Type: DS-CDMA
  - Modulation BW: 4.096 MHz
  - Channel BW: 5 MHz
  - Frequency Band: UHF 800 – 915 MHz
- DVB-T digital terrestrial television broadcasting
  - Modulation Type: OFDM
  - Modulation BW: 6.66 MHz

  - Channel BW: 7 MHz
  - Frequency Band: UHF 520 – 820 MHz (Broadcast Bands IV/V)

- DAB digital radio broadcasting (Transmission Mode I)
  - Modulation Type: OFDM
  - Modulation BW: 1.537 MHz
  - Channel BW: 1.75 MHz
  - Frequency Band: VHF 174 – 230 MHz (Broadcast Band III)

All analysis and performance testing carried out in later chapters will thus be presented in the context of these target applications. For example, a Class-AB push-pull power amplifier is chosen as part of the laboratory transmitter testbed (discussed in Chapter 4) specifically to replicate the distortion characteristics of these transmitters during algorithm testing.

It is also noted that the specific transmission frequencies and bandwidths listed above are based on the Australian Communications & Media Authority (ACMA) RF spectrum plan [13, 14] and licensed provider allocation within Australia [7, 15].

3.3 Assumptions

Assumptions Relating To The Reader of This Thesis

It is assumed that the reader of this thesis has a background in RF electronics, communications and signal processing. It is also assumed that the reader is familiar with object oriented programming, particularly the C^{++} language, and has access to Microsoft Visual Studio (or an equivalent IDE text editor) for viewing and navigating software source code. The reader is also encouraged to peruse the accompanying DVD as it contains material referenced within this thesis.

Assumptions Relating To The Writing of This Thesis

Software source code associated with this research will not be collated in an appendix of this thesis. Instead, it will be collated electronically on the accompanying DVD within the top level *Software* directory and referenced appropriately. The reason being, source code is written with horizontal span greater than that of A4 portrait, courtesy of the generous span of the IDE text editor. Blindly copying this code across to an A4 portrait page ultimately leads to line breaking and therefore fragmentation of the original formatting. Source code should be viewed electronically, with proper formatting, using a text editor.

It is also noted that in cases where a lengthy mathematical derivation would disrupt the flow of ideas at a particular point in the thesis, that mathematical derivation is forced to Appendix B and only the final result (plus reference to the Appendix) is inserted within the thesis.

Assumptions Relating To Algorithm Testing

In this research, algorithm testing is performed on a transmitter testbed assembled within the laboratory. As will be discussed in Chapter 4, the components of this testbed replicate the characteristics of a commercial transmitter and therefore testing results are highly relevant.

3.4 Delimitations

Delimitations Relating To Measures of Nonlinearity

The OFDM and DS-CDMA signals used in the target applications of Section 3.2 are continuous-spectra signals with substantial bandwidth. For this reason, the following measures of amplifier nonlinearity, applicable only to single-tone, two-tone or narrowband signals, will not be discussed:

- Compression
- Intercept Points
- AM-AM/AM-PM

This thesis will use the following, more appropriate measures of amplifier nonlinearity:

- Adjacent Channel Power Ratio (ACPR)
- Co-Channel Power Ratio (CCPR)

Delimitations Relating To Implementation

The focus of this research is on predistortion filter architecture and parameter estimation. Optimal implementation of these components in software / firmware / hardware, for the purpose of minimizing computation or memory requirements, will not be considered.

Delimitations Relating To Transmitter Channels

This research only considers single-channel radio transmitters. This is in contrast to multi-channel radio transmitters which combine multiple exciter outputs, each

with a different channel or carrier frequency, prior to the power amplifier stage. In addition, frequency agile channels and channel switching will not be considered. Once a service channel has been allocated, that channel will remain fixed.

3.5 Commercialization

One clearly defined output from this research activity is Intellectual Property and the potential for commercialization. A *Provisional* and *International* patent application entitled *A method and system for linearising a radio frequency transmitter* was filed on 04 January 2011 and 23 December 2011 respectively. This was followed by an *International Search Report* being generated on 12 July 2012. Commercialization opportunities are currently being sought with the help of commercialization company UniQuest (www.uniquest.com.au). Given its length, the full document set representing the patent description, drawings, claims, application filings and search report is forced to the accompanying DVD under the top level *Patent* folder. The summarizing *Claims* section is however included in Appendix C of this thesis.

3.6 Publications

Publications are another clearly defined output of this research activity. Two journal articles have been published in the IEEE Transactions on Broadcasting. The first article entitled *Adaptive Digital Predistortion for Wideband High Crest Factor Applications Based on the WACP Optimization Objective: A Conceptual Overview* outlines the concepts of the proposed method of digital predistortion and provides preliminary results. The second article entitled *Adaptive Digital Predistortion for Wideband High Crest Factor Applications Based on the WACP Optimization Objective: An Extended Analysis* is a more indepth presentation of the proposed method, discussing implementation aspects and providing further adaption results. Both publications are included in Appendix D of this thesis.

Chapter 4

Laboratory Transmitter Testbed

A radio transmitter testbed assembled in the laboratory is used to develop and test all algorithms associated with this research. A block diagram and photo of this testbed is presented in Figures 4.1 and 4.2 respectively. It is noted that in both figures, testbed components are labeled uniquely with a red letter in order to facilitate component matching across figures.

A Rohde & Schwarz AMIQ IQ Modulation Generator (C) implements the digital-to-analog conversion and reconstruction filtering processes whilst a Rohde & Schwarz SMIQ Vector Signal Generator (D) implements the IQ modulation and frequency upconversion processes. Power amplification is performed by a solid state, 25 Watt Class-AB push-pull power amplifier (G). Preceding this power amplifier is a driver amplifier (E). The output of the power amplifier is subsequently fed to a dummy load (K). A Rohde & Schwarz FSIQ Spectrum Analyzer (L) measures transmitter output Power Spectral Density via a directional coupler (J) at the output of the power amplifier. Signal encoding and modulation (A), predistortion filtering (B), objective function measurement (M) and mathematical optimization (N) are all implemented in software running on a standalone PC (O) with GPIB capability. The user is able to externally configure, control and monitor the testbed via a DOS shell console interface (P) offering numerous built in menu options. All software is written in C^{++} with object-oriented design [123, 267]. This software and all test instruments communicate via GPIB. A further discussion of these testbed components is given in the following.

4.1 AMIQ IQ Modulation Generator (C)

Technical specifications of the Rohde & Schwarz AMIQ IQ Modulation Generator (Model: 1110.2003.04) are given in [229, 230]. The instrument consists of two channels (IQ) of SDRAM, digital-to-analog conversion and reconstruction filtering.

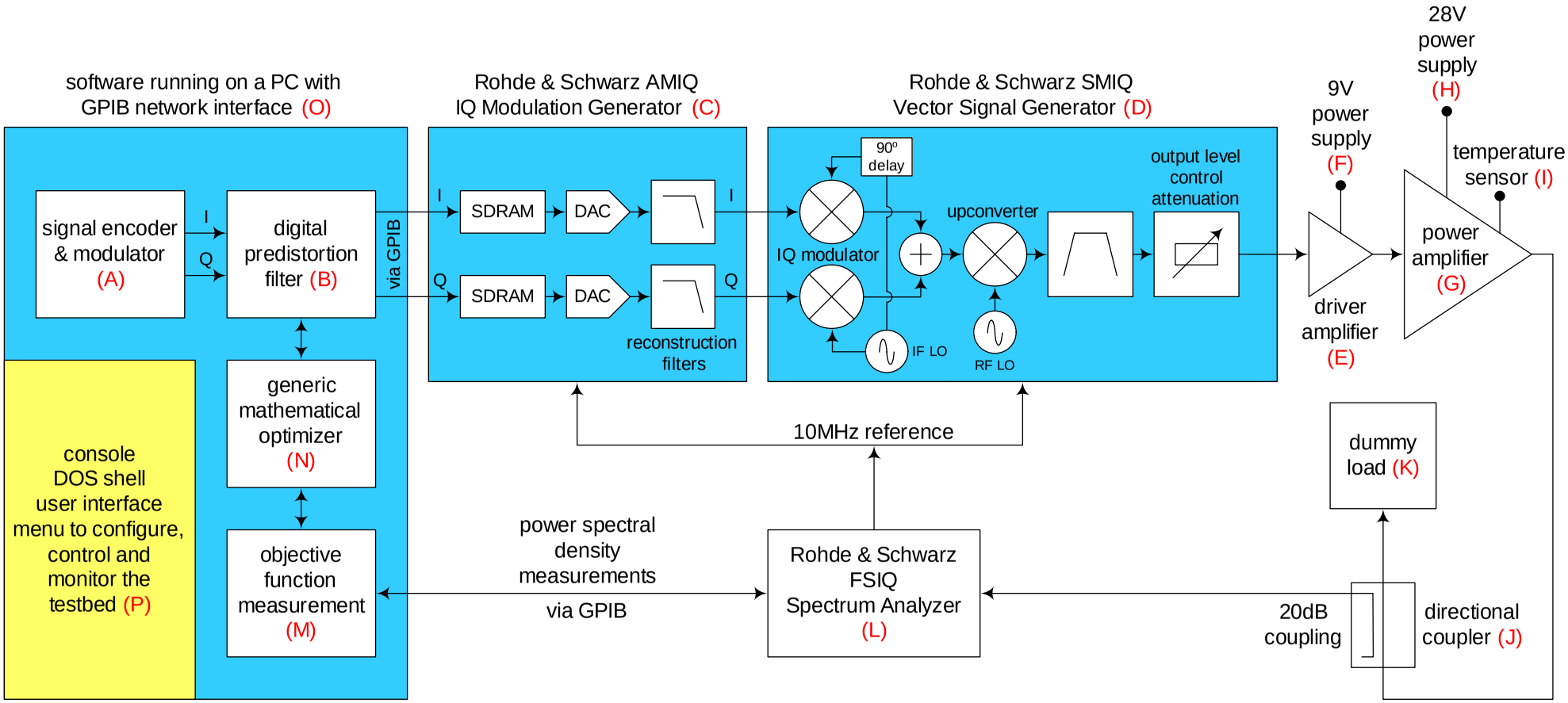


Figure 4.1: Block diagram of the laboratory transmitter testbed

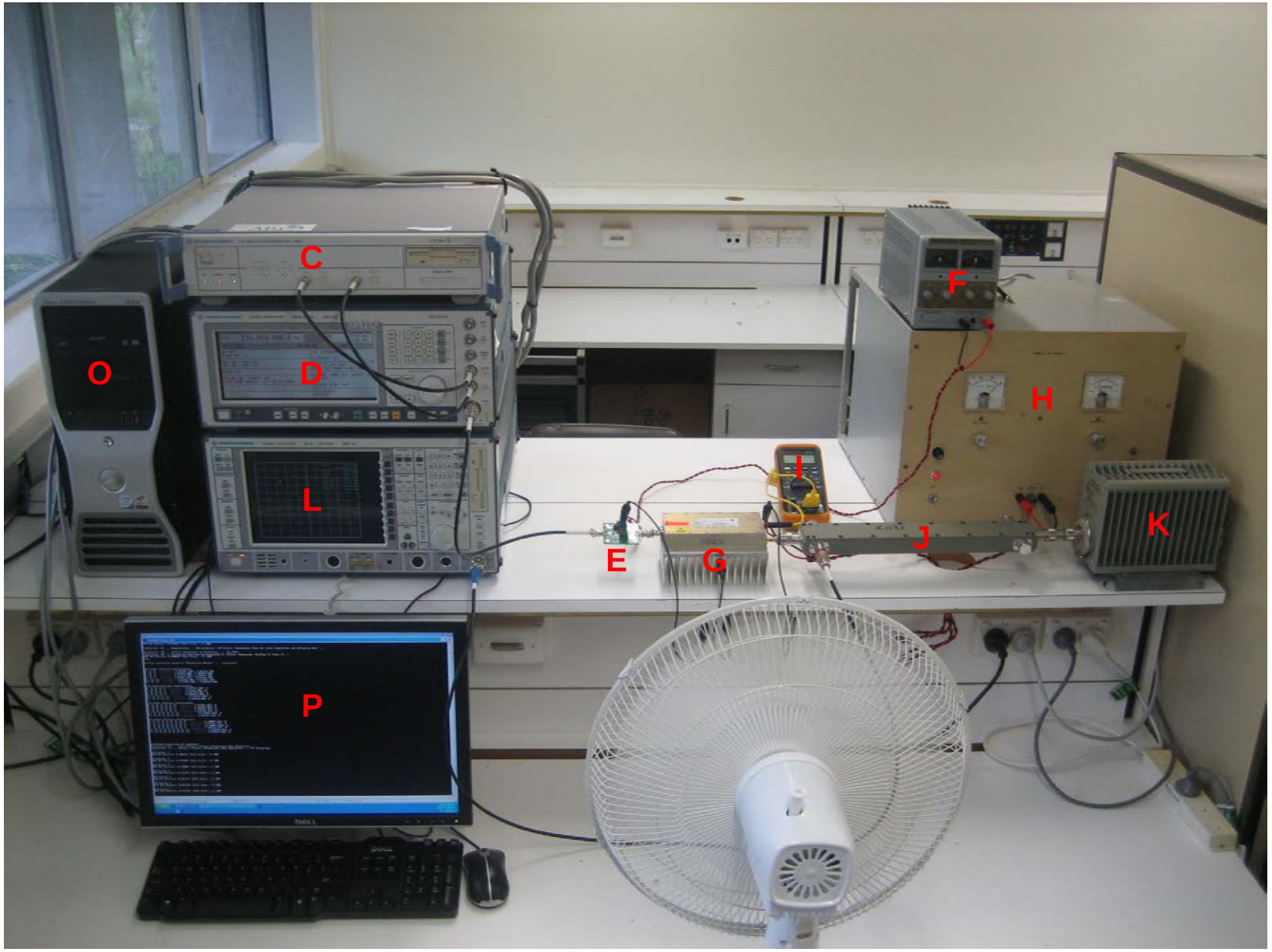


Figure 4.2: Photo of the laboratory transmitter testbed

The complex predistortion filter output signal is transfered to the instrument's SDRAM via GPIB and clocked to the DACs at the preset sampling rate. The AMIQ's output clocking frequency can be set in the range 10-100 MHz while its reconstruction filters can be set to either a 2.5 MHz or 25 MHz cutoff frequency. Knowing that the predistortion filter will generate high order spectral regrowth, there is no choice but to set the reconstruction filters to the maximum 25 MHz cut-off frequency for each of the target applications. The following sampling frequencies (within the 100 MHz maximum limit) are chosen for each of the target applications:

- DAB: 65.536 MHz
- WCDMA: 92.160 MHz
- DVB-T: 64 MHz

Two points are worth clarifying with respect to the DVB-T signal:

1. The sampling rate of the DVB-T signal is lower than that of the DAB and WCDMA signals despite exhibiting a greater continuous-time bandwidth. This is because the next power of two IFFT would bring the sampling rate to greater than the 100 MHz maximum. The current sampling rate is satisfactory however as it oversamples the DVB-T signal by at least the highest predistortion filter nonlinearity (9^{th} order as will become evident in Chapter 8) and therefore avoids spectral regrowth aliasing.

2. Theoretically, the 25 MHz reconstruction filters can only fully reconstruct up to the 7^{th} order components of the predistorted DVB-T signal. In practice this is not a problem however as the tailing end of the 9^{th} order predistortion spectrum situated beyond the reconstruction filter cutoff frequency (and therefore linearly distorted) will not have any effect on the final predistortion process as it will fall beneath the spectrum analyzer noise floor. In conclusion, the useful portion of the 9^{th} order predistortion spectrum will be adequately reconstructed.

In this transmitter testbed, a scaling mechanism is built into the software between the predistortion filter output and AMIQ input, to control and monitor how much of the DAC's full scale range is being utilized at any specific time. Prior to predistortion, this scaling mechanism is set such that peak signal excursion is approximately 30% of full scale. This is for good reason. During the predistortion process, signal peaks expand to compensate for the compression effects of the power amplifier and as a result, peak signal excursion grows from the intial 30% to approximately 80% of full scale. If this growth in peak signal excursion wasn't taken into account in the initial setting of the scaling mechanism, the DACs would end up

nonlinearly distorting and clipping the signal during predistortion, leading to major spurious components.

As discussed in Chapter 1, a downside to using these OFDM / CDMA signals is the need to accommodate their large Crest Factors during physical transmission. This applies not only to the power amplifier but also to the DACs. With the scaling mechanism discussed above set to accommodate the signal peaks, the *average* signal excursion is significantly less than 30% of full scale due to the high Crest Factors involved[1]. With only a small fraction of the full scale range therefore being utilized on average, the finite resolution limitations of the DACs become apparent. In general, broadband quantisation error can be expected, leading to a raised spectral noise floor and therefore reduced spectral dynamic range. In the AMIQ, DACs possess 14 bits of resolution leading to a spectral dynamic range of approximately 50 dBc.

4.2 SMIQ Vector Signal Generator (D)

Technical specifications of the Rohde & Schwarz SMIQ Vector Signal Generator (Model: SMIQ06B 1125.5555.06) are given in [232,233]. The instrument is operated as a standard IQ modulator and RF upconverter (no additional model features included) with the following non-default settings to ensure greater output linearity and level repeatability:

- Output Mode - Low Distortion
- Automatic Level Control - Off - Table Mode

Recalibration of the level control table is performed regularly as outlined in the application note [19]. The instrument's output power is controlled by a level setting which internally configures the output attenuation. The maximum linear output power of the instrument is not enough to drive the power amplifier into severe nonlinear operation hence the need for a driver amplifier (E) applied at the instrument's output. It follows that the instrument's level setting is used to indirectly control the input drive level of the power amplifier.

4.3 Driver Amplifier (E)

The driver amplifier used to boost SMIQ output power is the Mini-Circuits Gali-52. Technical specifications can be found in the data sheet included in Appendix E. This is a wideband amplifier with approximately 20 dB of gain and a 15.5 dBm output power rating. The actual output power necessary to drive the power amplifier into

[1]Expansion of signal peaks to 80% of full scale during predistortion has little effect on the average power.

saturation is approximately -3 dBm. This is 18.5 dB below the output power rating and hence the driver amplifier operates linearly.

4.4 Power Amplifier (G)

The power amplifier used in the laboratory transmitter testbed is the OPHIR RF 5303038. This power amplifier's characteristics are consistent with those of our target application power amplifier modules and therefore it can be expected to replicate their nonlinear behavior in a laboratory test environment. It is a solid state, broadband, 25 Watt Class-AB push-pull power amplifier. Technical specifications can be found in the data sheet included in Appendix E. Transmitter power amplifier modules operate in Class-AB push-pull to obtain the best tradeoff between efficiency and linearity. In light of reduced manufacturing cost and spares inventory, they are also broadband in nature, avoiding channelized designs [40, 194, 237].

Fan forced cooling is applied to the power amplifier (as can be seen in Figure 4.2 on Page 34) with temperature monitored via a thermocouple (I) attached to the amplifier's casing. Amplifier temperature is controlled via the fan speed and therefore airflow over the casing. A greater airflow reduces operating temperature and vice versa. This simple but effective means of manipulating amplifier temperature is used in Section 14.2 to vary the amplifier's nonlinear transfer characteristic and therefore to test algorithm adaptivity.

4.5 FSIQ Spectrum Analyzer (L)

Technical specifications of the Rohde & Schwarz FSIQ Spectrum Analyzer (Model: FSIQ26 1119.6001.27) are given in [231]. It is noted that this instrument is technically referred to as a *signal analyzer* as it has two modes of operation; a spectrum analyzer mode and a vector analyzer mode. The vector analyzer mode allows the analysis of digital modulations in the constellation-domain, however this mode will not be used. The instrument is strictly set to spectrum analyzer mode and will therefore be referred to as such.

The spectrum analyzer is configured such that the input attenuation is coupled to the reference level. In light of the signal's initially high and continually growing Crest Factor (growth occurs during predistortion), the reference level must be set higher than normal to ensure the input mixer is not overdriven. A further safe guard against overdriving the input mixer is the non-default setting:

- Attenuation Auto Mode - Low Distortion

which adds an additional 10 dB to the auto coupled input attenuation.

It is worth noting that for high Crest Factor signals, maximum theoretical dynamic range is actually achieved when the input attenuation is decoupled from the reference level (opposite to above) and set such that the peak input signal power is approximately 10 dB below the 1 dB compression point of the signal path prior to the IF filter [287]. However in practice, decoupling of the input attenuation is unnecessary and will not provide any extra performance in our case as it is the AMIQ's substantial quantization noise which determines the overall dynamic range.

To obtain stable and repeatable power spectrum measurements, an RMS detector is selected [287]. Resolution bandwidth is set to 30 KHz (0.45% to 2% of signal bandwidth), resulting in adequate frequency resolution and a practical 2 second sweep time [225]. Video bandwidth is set to 10 times the resolution bandwidth, thereby avoiding averaging and allowing the true power spectrum to be measured [72].

4.6 Signal Encoding and Modulation (A)

The signal encoding and modulation process is encapsulated in software as an object-oriented class [249]. Given the slightly different encoding and modulation requirements, a separate class is defined for DAB, DVB-T and WCDMA. During software build, conditional compilation directives (#if, #elif) allow the developer to select which one of these classes to instantiate as the testbed's encoder/modulator object. Class data members include:

- the modulated signal:
  - that which is applied to the predistortion filter
  - stored as a complex vector
- parameters of the modulated signal:
  - number of OFDM carriers (DAB and DVB-T classes)
  - guard interval (DAB and DVB-T classes)
  - flag to remove carriers and monitor CCD (DAB and DVB-T classes)
  - encoding type (DAB, DVB-T and WCDMA classes)
  - chipping rate (WCDMA class)
  - spreading factor (WCDMA class)
  - user channels (WCDMA class)
  - RRC rolloff factor (WCDMA class)

Class member functions include:

- generating the modulated signal
- writing the modulated signal to file
- reading the modulated signal from file
- computing modulated signal characteristics such as Crest Factor and peak power
- notch filtering the modulated signal to monitor CCD (WCDMA class only)

The DAB class is configured to encode and modulate according to the DAB digital radio broadcasting standard (Transmission Mode I) as follows:

- Modulation Type: OFDM
- Modulation BW: 1.537 MHz
- Channel BW: 1.75 MHz
- Encoding: QPSK
- Symbol Duration: 1 ms
- Guard Interval: 246 μs
- Carriers: 1536 spaced 1KHz apart

The DVB-T class is configured to encode and modulate according to the DVB-T digital terrestrial television broadcasting standard (either 8K or 2K Mode) as follows:

- Modulation Type: OFDM
- Modulation BW: 6.66 MHz
- Channel BW: 7 MHz
- Encoding: 16 or 64 QAM
- Symbol Duration:
  - 8K Mode: 1024 μs
  - 2K Mode: 256 μs
- Guard Interval: 1/4, 1/8, 1/16 or 1/32 of symbol duration

- Carriers:
  - 8K Mode: 6817 spaced 976.5625 Hz apart
  - 2K Mode: 1705 spaced 3.90625 KHz apart

The WCDMA class is configured to encode and modulate according to the 3G UMTS WCDMA standard as follows:

- Modulation Type: DS-CDMA
- Modulation BW: 4.096 MHz
- Channel BW: 5 MHz
- Encoding: QPSK
- Chipping Rate: 3.84 Mchips/s
- RRC Rolloff Factor: 0.22
- User Channels: 512

It is noted that the number of WCDMA user channels above (512) is set higher than the standard 256. This is done intentionally to increase the signal's Crest Factor and hence create a more difficult test scenario [122, 227].

A plot of the Complementary Cumulative Distribution Function (CCDF) representative of a DVB-T, WCDMA and DAB signal taken over a finite 1 000 000 signal sample duration is presented in Figure 4.3. Also included for comparison are the CCDF plots for FM modulation, QPSK and 16 QAM. These CCDF plots are derived using the envelope power approach [18]. The CCDF plot of a signal represents the probability (y-axis) of the signal's instantaneous power exceeding its mean power by a specified value (x-axis) [235]. The Crest Factor of a signal can subsequently be derived statistically from its CCDF plot as the x-axis intercept point [234]. The Crest Factor derived this way clarifies Table 1.1's Crest Factor values.

4.7 Predistortion Filtering (B), Objective Function Measurement (M) and Mathematical Optimization (N)

Since these testbed components are the focus of our research endeavor, their details and workings will be developed throughout the remainder of this thesis. At this stage however, it is sufficient to understand that these components are implemented in software running on the standalone PC (O) and configured via the console interface (P).

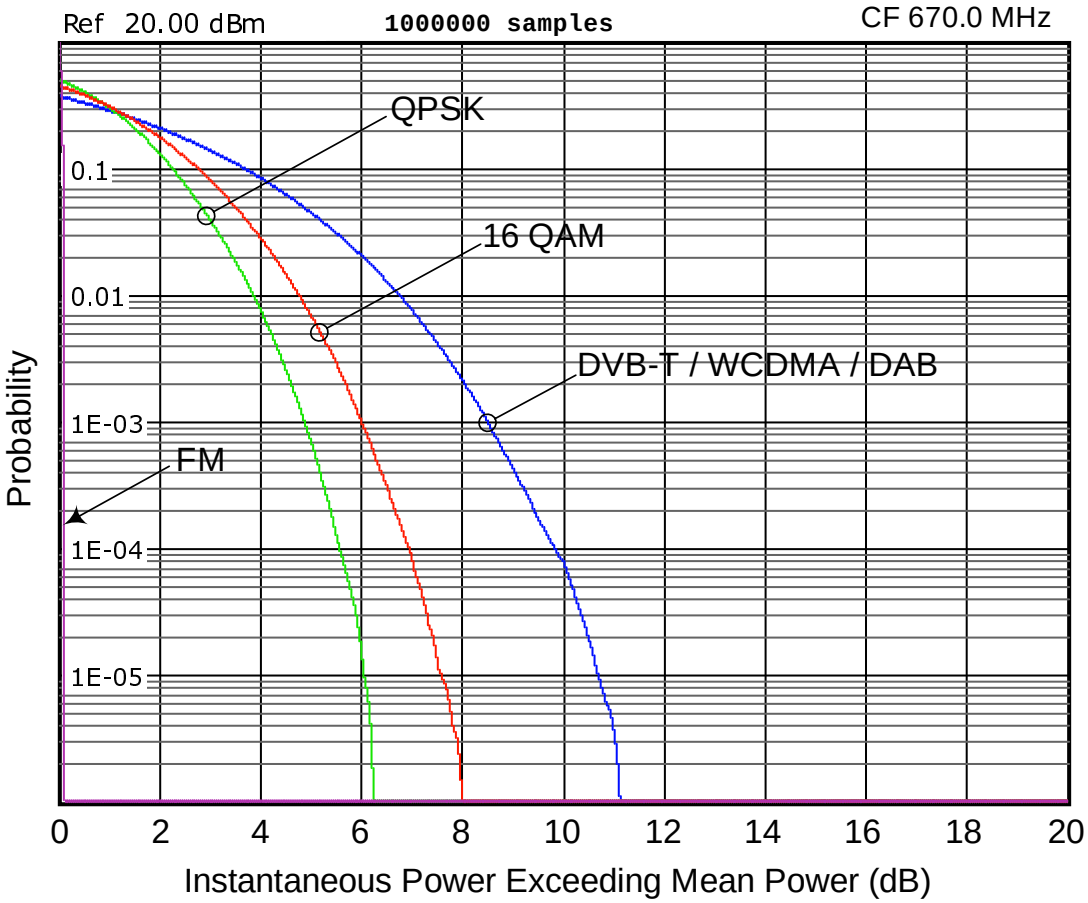


Figure 4.3: Complementary Cumulative Distribution Functions

4.8 Console Interface (P)

The transmitter testbed is configured and controlled via a console DOS shell user interface menu. After system initialization, program execution enters a repetition and selection structure which implements the user interface menu. The available menu options allow the user to:

- Measure ACP, CCDF and Crest Factor data
- Manipulate spectrum analyzer traces to log past / present ACP reduction
- Set the AMIQ DAC scaling mechanism
- Enable / disable individual predistortion filter nonlinearities
- Configure predistortion filter coefficients (reset / load / save / display / increment)
- Configure optimization objective function characteristics

- Generate / load / save encoded and modulated signals
- Run individual or sets of optimization algorithms in the form of a schedule
- View testbed setup details

The console interface also provides real-time textual and numeric feedback, allowing the user to monitor the state of the testbed and its predistortion algorithms at all times. It follows that with this console interface, the researcher is able to configure, control and monitor all the experiments involved in algorithm development and testing.

With the first four chapters of this thesis devoted to introducing the research endeavor and testing environment, we now turn attention towards accomplishing our specific statement of research.

Chapter 5

Volterra Series Modeling of Amplifier & Predistorter

In this chapter, we introduce the nonlinear Volterra Series and discuss its intermodulating and spectral regrowth properties. With the RF power amplifier then modeled as such, we convert the RF transmitter model to its baseband equivalent thereby showing that the resulting predistortion filter takes on a variant architecture of the pure Volterra Series called the *Baseband* Volterra Series.

As discussed in the *Literature Review* of Chapter 2, those predistortion filter architectures proposed for today's wideband applications possess some form of memory and are behavioral models rather than physical circuit level representations. Memory is required to compensate for the dynamics of the power amplifier (now modulated with wideband signals) while behavioral models are favored due to their lower complexity and processing requirements. Conventional behavioral models with memory include the Volterra Series, Memory Polynomial, NARMA filter, Hammerstein and Wiener filters, TNTB model as well as variants and hybrids of each. The reader is directed to page 14 for further details of these models.

In this research, the Volterra Series is chosen as the foundation model for the power amplifier and predistortion filter for the following high level reasons:

Model Generality - Of all models, the Volterra Series is the most general. If the Volterra Series is not capable of representing a nonlinear system, then no other model will [280, 289].

Mathematical Tractability - Of all models, the Volterra Series is the most mathematically tractable. Compared to other models, it provides the greatest insight into nonlinear system interaction and behavior [117, 278].

Application to Future Wider Band Systems - Of all models, the Volterra Series exhibits the greatest degrees of freedom in memory thus offering the greatest potential for compensating those complex dynamic nonlinearities expected of future wider band systems [209, 274]. It is anticipated that all other models will become inadequate with growth in bandwidth.

It is understood that any foundation Volterra Series model will have a large kernel size and some form of pruning will be necessary for final implementation of the predistortion filter. In Chapter 12 we address this pruning aspect. Prior to Chapter 12 however, we specifically ignore pruning in order to maximize insight into algorithm development. That is, in terms of nonlinear system modeling, we favor the idea of staying as general as possible for as long as possible, only specializing (in this case pruning) when needed. This allows our work to develop freely without fear of being pushed in a specific direction based on the limitations of any specific model. The benefit of this guiding mindset is that our work will remain applicable to a broader range of future wideband standards. That is, as will become apparent in later chapters, the pruning strategy will be the only part of the proposed technique needing to be matched to a varying bandwidth modulation standard. In this sense, the pruning strategy can be seen to act as the binding link between a constant technique and the forever changing application space.

5.1 Introduction To The Volterra Series

The Volterra Series was first studied by mathematician Vito Volterra [276, 277]. It is suitable for modeling mildly nonlinear[1] time invariant systems and can be represented in either *operator* or *functional* form. In conjunction with Figure 5.1, the *operator* form is given by:

$$y(t) = \boldsymbol{H}[x(t)] = \sum_{n=1}^{\infty} \boldsymbol{H}_n[x(t)] \tag{5.1}$$

where $\boldsymbol{H}[\,\cdot\,]$ represents the *system* Volterra operator, $x(t)$ and $y(t)$ represent system input and output respectively and $\boldsymbol{H}_n[\,\cdot\,]$ represents the n^{th} order Volterra operator, that is, the *individual* n^{th} order nonlinearity.
The continuous-time, causal, pure Volterra Series in *functional* form is given by:

$$y(t) = \sum_{n=1}^{\infty} \left(\int_0^{\infty} \cdots \int_0^{\infty} h_n(\tau_1, \ldots, \tau_n)\, x(t-\tau_1) \cdots x(t-\tau_n)\, d\tau_1 \cdots d\tau_n \right) \tag{5.2}$$

[1] $3^{\text{rd}} \gg 5^{\text{th}} \gg 7^{\text{th}}$ order distortion which is consistent with our intended application

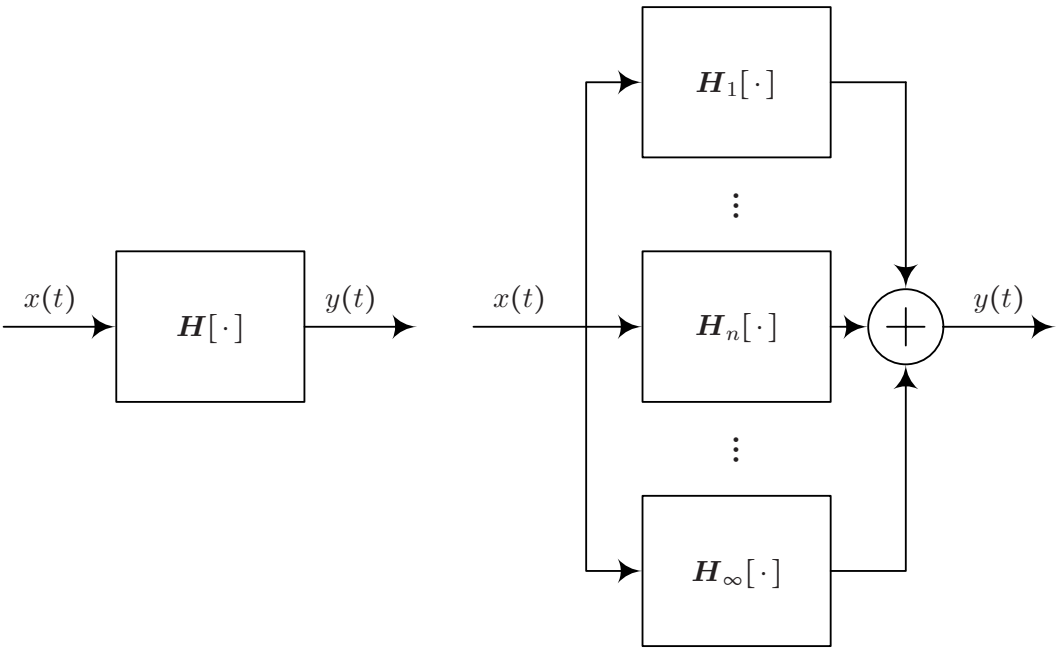


Figure 5.1: Operator form of Volterra Series

where $h_n(\tau_1, \ldots, \tau_n)$ represents the n^{th} order Volterra kernel. The entire set of kernels ($n = 1$ to ∞) fully characterizes the nonlinear Volterra system. Kernels are real in general and are functions of input memory. Equating (5.1) and (5.2) gives the *functional* form of the n^{th} order Volterra operator:

$$\boldsymbol{H}_n[x(t)] = \int_0^\infty \cdots \int_0^\infty h_n(\tau_1, \ldots, \tau_n)\, x(t-\tau_1) \cdots x(t-\tau_n)\, d\tau_1 \cdots d\tau_n \tag{5.3}$$

In our analysis, kernel symmetry is assumed, that is $h_n(\tau_1, \ldots, \tau_n) = h_n(\tau_n, \ldots, \tau_1)$, without loss of generality since any nonsymmetric kernel can be symmetrized [240, 248]. To understand the nonlinear Volterra interaction between input signal components, let the input $x(t)$ now be represented as a general *sum of weighted signal components*:

$$x(t) = \sum_{a=1}^{A} c_a s_a(t) \quad \text{for } A \geq 1 \tag{5.4}$$

The response of the n^{th} order Volterra operator then becomes:

$$\boldsymbol{H}_n\left[\sum_{a=1}^{A} c_a s_a(t)\right] = \sum_{a_1=1}^{A} \cdots \sum_{a_n=1}^{A} c_{a_1} \cdots c_{a_n} \left(\int_0^\infty \cdots \int_0^\infty h_n(\tau_1, \ldots, \tau_n)\, s_{a_1}(t-\tau_1) \cdots s_{a_n}(t-\tau_n)\, d\tau_1 \cdots d\tau_n \right) \tag{5.5}$$

The right hand bracketed term in (5.5) is formally known in the literature as the n-linear Volterra operator and represented as:

$$\boldsymbol{H}_n\big\{s_{a_1}(t),\ldots,s_{a_n}(t)\big\} = \int_0^\infty \cdots \int_0^\infty h_n(\tau_1,\ldots,\tau_n)\; s_{a_1}(t-\tau_1)\cdots s_{a_n}(t-\tau_n)\; d\tau_1\cdots d\tau_n \tag{5.6}$$

It is given this name since it is linear in each argument when all others are held fixed. With this formal representation, the response of the n^{th} order Volterra operator becomes:

$$\boldsymbol{H}_n\Big[\sum_{a=1}^{A} c_a s_a(t)\Big] = \sum_{a_1=1}^{A}\cdots\sum_{a_n=1}^{A} c_{a_1}\cdots c_{a_n}\; \boldsymbol{H}_n\big\{s_{a_1}(t),\ldots,s_{a_n}(t)\big\} \tag{5.7}$$

From this expression, we can make some very important observations:

1. Unlike a linear system where the response to a *sum of weighted signal components* would be a *sum of weighted component responses*, we now have interaction between the different input signal components, leading to output *inter-modulation* responses.

2. Since multiplication in the time-domain is equivalent to convolution in the frequency-domain, specifically referring to the product terms within the integral of the n-linear Volterra operator (5.6), output spectral regrowth is generated by the *inter-modulating* signal components.

To investigate this spectral regrowth further, we turn to a frequency-domain representation of the Volterra Series. Let the input $x(t)$ now be represented as a general *accumulation of weighted spectral components*:

$$x(t) = \int_{-\infty}^{\infty} X(f)\; e^{j2\pi ft}\; df \tag{5.8}$$

The response of the n^{th} order Volterra Series operator to this input is:

$$\boldsymbol{H}_n\Big[\int_{-\infty}^{\infty} X(f)\; e^{j2\pi ft}\; df\Big] = \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} X(f_1)\cdots X(f_n)\; e^{j2\pi(f_1+\cdots+f_n)t} \left(\int_0^\infty\cdots\int_0^\infty h_n(\tau_1,\ldots,\tau_n)\; e^{-j2\pi(f_1\tau_1+\cdots+f_n\tau_n)}\; d\tau_1\cdots d\tau_n\right) df_1\cdots df_n \tag{5.9}$$

The right hand bracketed term in (5.9) is formally known in the literature as the n-dimensional Fourier Transform of the n^{th} order Volterra kernel and expressed as:

$$H_n(f_1,\ldots,f_n) = \int_0^\infty \cdots \int_0^\infty h_n(\tau_1,\ldots,\tau_n)\, e^{-j2\pi(f_1\tau_1+\cdots+f_n\tau_n)}\, d\tau_1\cdots d\tau_n \tag{5.10}$$

Substituting (5.10) into (5.9) gives the response in terms of input signal components:

$$\boldsymbol{H}_n\left[\int_{-\infty}^{\infty} X(f)\, e^{j2\pi ft}\, df\right] = \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} X(f_1)\cdots X(f_n)\, H_n(f_1,\ldots,f_n)\, e^{j2\pi(f_1+\cdots+f_n)t}\, df_1\cdots df_n \tag{5.11}$$

In terms of spectral regrowth, (5.11) tells us that input spectral components existing at $f_1,\ldots,f_n$ will generate an output spectral component at $f_1+\ldots+f_n$ with complex amplitude:

$$Y_n(f_1,\ldots,f_n) = X(f_1)\cdots X(f_n)\, H_n(f_1,\ldots,f_n) \tag{5.12}$$

It is important to highlight that $Y_n(f_1,\ldots,f_n)$ in (5.12) represents the output contribution from only one set of n inter-modulating spectral components. Since many different sets $[f_1,\ldots,f_n]$ exist for which $f_1+\ldots+f_n = f$, the *resultant* output spectral component at frequency f is obtained by accumulating all such individual contributions:

$$Y_n(f) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} Y_n(f-\varphi_1,\ \varphi_1-\varphi_2,\ldots,\varphi_{n-2}-\varphi_{n-1},\ \varphi_{n-1})\, d\varphi_1\, d\varphi_2\cdots d\varphi_{n-1} \tag{5.13}$$

It follows that (5.13) is formally known in the literature as the Fourier Transform of the output of the n^{th} order Volterra operator.

The above discussion has introduced the Volterra Series and its inter-modulating properties in the most general sense. In the next section, we apply this Volterra model to the power amplifier at RF and subsequently derive an equivalent baseband transmitter model.

5.2 RF and Baseband Transmitter Models

Figure 5.2 presents a block diagram of the transmitter as it would exist in physical implementation. We refer to this as an *RF transmitter model* since the power amplifier processes an RF signal. In our predistortion work however, we desire an equivalent transmitter model in which the power amplifier processes a baseband

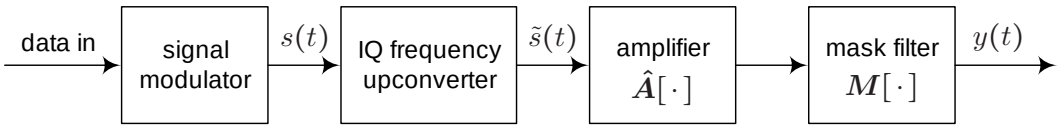


Figure 5.2: RF transmitter model

signal. The reason being, the mathematical architecture of the power amplifier in this equivalent *baseband transmitter model* dictates the mathematical architecture of the corresponding baseband predistortion filter. In the following, we derive this equivalent baseband transmitter model and hence present the required predistortion filter architecture.[2]

Referring to Figure 5.2, the output of the signal modulator is represented by the complex baseband signal $s(t)$. Following IQ frequency upconversion[3], the real RF excitation signal $\tilde{s}(t)$ can be represented as:

$$\tilde{s}(t) = s(t)\, e^{j2\pi ft} + s^*(t)\, e^{-j2\pi ft} \tag{5.14}$$

where f represents the transmission carrier frequency and * denotes complex conjugation. In this form, $s(t)$ is commonly referred to as the *complex baseband signal envelope*. In order to simplify notation, let:

$$\tilde{s}_{+1}(t) = s(t)\, e^{j2\pi ft} \quad \text{and} \quad \tilde{s}_{-1}(t) = s^*(t)\, e^{-j2\pi ft} \tag{5.15}$$

Substituting back into (5.14), $\tilde{s}(t)$ can now be represented as:

$$\tilde{s}(t) = \tilde{s}_{+1}(t) + \tilde{s}_{-1}(t) \tag{5.16}$$

Here, the $+1$ and -1 subscripts are intended to signify the positive and negative bandpass frequency characteristics of the respective signal components.

Representing the amplifier and mask filter in terms of the operators $\hat{\boldsymbol{A}}[\,\cdot\,]$ and $\boldsymbol{M}[\,\cdot\,]$ respectively, the transmitter output $y(t)$ can be written as:

$$y(t) = \boldsymbol{M}\Big[\, \hat{\boldsymbol{A}}\big[\, \tilde{s}(t) \,\big] \,\Big] \tag{5.17}$$

Substituting (5.16) into (5.17) then gives:

$$y(t) = \boldsymbol{M}\Big[\, \hat{\boldsymbol{A}}\big[\, \tilde{s}_{+1}(t) + \tilde{s}_{-1}(t) \,\big] \,\Big] \tag{5.18}$$

[2]The power amplifier modeling community also shares this desire for baseband modeling but for different reasons. In their case, a baseband transmitter model avoids the problems associated with high carrier frequency sampling rates [39].

[3]Ideal frequency upconversion is assumed here since mixer nonlinearities are considered negligible compared to those nonlinearities of the power amplifier.

Letting the amplifier $\hat{\boldsymbol{A}}[\,\cdot\,]$ be represented as a Volterra system, (5.18) can be expressed in terms of individual n^{th} order Volterra operators:

$$y(t) = \boldsymbol{M}\Bigg[\ \sum_{n=1}^{\infty} \hat{\boldsymbol{A}}_n\big[\ \tilde{s}_{+1}(t) + \tilde{s}_{-1}(t)\ \big]\ \Bigg] \tag{5.19}$$

Expressing further in terms of n-linear Volterra operators:

$$y(t) = \boldsymbol{M}\Bigg[\ \sum_{n=1}^{\infty} \Bigg(\sum_{a_1=\pm1} \cdots \sum_{a_n=\pm1} \hat{\boldsymbol{A}}_n\Big\{\tilde{s}_{a_1}(t), \ldots, \tilde{s}_{a_n}(t)\Big\} \Bigg)\ \Bigg] \tag{5.20}$$

Since the bandpass mask filter $\boldsymbol{M}[\,\cdot\,]$ limits spectral regrowth to the *first-zone* carrier region, only those $\hat{\boldsymbol{A}}_n\{\tilde{s}_{a_1}(t), \ldots, \tilde{s}_{a_n}(t)\}$ terms in (5.20) for which $(a_1 + \cdots + a_n = \pm 1)$ will actually be transmitted. If we let $\boldsymbol{a}_{+1}$ and $\boldsymbol{a}_{-1}$ represent the vector subspaces of $\big\{\,[a_1, \cdots, a_n]\,\big\}$ for which $(a_1 + \cdots + a_n = +1)$ and $(a_1 + \cdots + a_n = -1)$ respectively, then (5.20) can be rewritten as:

$$y(t) = \boldsymbol{M}\Bigg[\ \sum_{\text{odd } n=1}^{\infty} \Bigg(\sum_{\boldsymbol{a}_{+1}} \hat{\boldsymbol{A}}_n\Big\{\tilde{s}_{a_1}(t), \ldots, \tilde{s}_{a_n}(t)\Big\} + \sum_{\boldsymbol{a}_{-1}} \hat{\boldsymbol{A}}_n\Big\{\tilde{s}_{a_1}(t), \ldots, \tilde{s}_{a_n}(t)\Big\} \Bigg)\ \Bigg] \tag{5.21}$$

It is noted that even order n terms no longer exist in the outer summation and hence the bandwidth of spectral regrowth components within the *first-zone* carrier region must be odd multiples of the original modulation bandwidth. Assuming now symmetric Volterra kernels, the inner summations of (5.21) can be simplified in terms of binomial coefficients:

$$\begin{aligned} y(t) = \boldsymbol{M}\Bigg[\ \sum_{\text{odd } n=1}^{\infty} & \binom{n}{\lceil \frac{n}{2} \rceil} \hat{\boldsymbol{A}}_n\Big\{\tilde{s}_{+1}(t)_1, \ldots, \tilde{s}_{+1}(t)_{\lceil \frac{n}{2} \rceil}, \tilde{s}_{-1}(t)_1, \ldots, \tilde{s}_{-1}(t)_{\lfloor \frac{n}{2} \rfloor}\Big\} \\ & + \binom{n}{\lceil \frac{n}{2} \rceil} \hat{\boldsymbol{A}}_n\Big\{\tilde{s}_{-1}(t)_1, \ldots, \tilde{s}_{-1}(t)_{\lceil \frac{n}{2} \rceil}, \tilde{s}_{+1}(t)_1, \ldots, \tilde{s}_{+1}(t)_{\lfloor \frac{n}{2} \rfloor}\Big\}\ \Bigg] \end{aligned} \tag{5.22}$$

Expressing the amplifier n-linear Volterra operators in functional form and expanding the bandpass signals in terms of their complex baseband envelope according

to (5.15), then gives:

$$y(t) = \boldsymbol{M}\Bigg[\sum_{\text{odd } n=1}^{\infty} e^{j2\pi ft} \int_0^{\infty} \cdots \int_0^{\infty} a_n(\tau_1,\ldots,\tau_n)\, s(t-\tau_1)\cdots s(t-\tau_{\lceil \frac{n}{2} \rceil})\, s^*(t-\tau_{\lceil \frac{n}{2} \rceil+1})\cdots s^*(t-\tau_n)\, d\tau_1 \cdots d\tau_n + e^{-j2\pi ft} \int_0^{\infty} \cdots \int_0^{\infty} a_n^*(\tau_1,\ldots,\tau_n)\, s^*(t-\tau_1)\cdots s^*(t-\tau_{\lceil \frac{n}{2} \rceil})\, s(t-\tau_{\lceil \frac{n}{2} \rceil+1})\cdots s(t-\tau_n)\, d\tau_1 \cdots d\tau_n \Bigg] \tag{5.23}$$

where:

$$a_n(\tau_1,\ldots,\tau_n) = \binom{n}{\lceil \frac{n}{2} \rceil}\, \hat{a}_n(\tau_1,\ldots,\tau_n)\, e^{j2\pi f(-\tau_1-\cdots-\tau_{\lceil \frac{n}{2} \rceil}+\tau_{\lceil \frac{n}{2} \rceil+1}+\cdots+\tau_n)} \tag{5.24}$$

and $\hat{a}_n(\tau_1,\ldots,\tau_n)$ is the amplifier n^{th} order Volterra kernel. To simplify (5.23), we define the *modified* amplifier system operator:

$$\boldsymbol{A}\big[s(t)\big] = \sum_{\text{odd } n=1}^{\infty} \boldsymbol{A}_n\big[s(t)\big] \tag{5.25}$$

where the associated individual n^{th} order nonlinear operators are defined as:

$$\boldsymbol{A}_n\big[s(t)\big] = \int_0^{\infty} \cdots \int_0^{\infty} a_n(\tau_1,\ldots,\tau_n)\, s(t-\tau_1)\cdots s(t-\tau_{\lceil \frac{n}{2} \rceil})\, s^*(t-\tau_{\lceil \frac{n}{2} \rceil+1})\cdots s^*(t-\tau_n)\, d\tau_1 \cdots d\tau_n \tag{5.26}$$

Substituting (5.26) back into (5.23) we obtain:

$$y(t) = \boldsymbol{M}\Bigg[\sum_{\text{odd } n=1}^{\infty} e^{j2\pi ft}\, \boldsymbol{A}_n\big[s(t)\big] + e^{-j2\pi ft}\, \boldsymbol{A}_n^*\big[s(t)\big] \Bigg] \tag{5.27}$$

Regrouping the summation terms according to (5.25) then gives:

$$y(t) = \boldsymbol{M}\Big[\boldsymbol{A}\big[s(t)\big]\, e^{j2\pi ft} + \boldsymbol{A}^*\big[s(t)\big]\, e^{-j2\pi ft} \Big] \tag{5.28}$$

We immediately recognize (5.28) as representing the *modified* amplifier system operator $\boldsymbol{A}[\,\cdot\,]$ acting on the pure baseband signal modulation $s(t)$ followed by an ideal IQ modulator and mask filter as depicted in Figure 5.3. Since the *modified* amplifier system operator is processing at baseband, this represents the baseband transmitter model we so desire. For now obvious reasons, the functional form of this *modified* amplifier system operator, (5.25) and (5.26), is known in the literature as

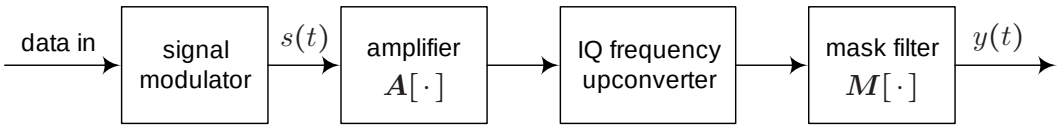


Figure 5.3: Baseband transmitter model

the *Baseband Volterra Series*. For the sake of consistency, we too will use this name from herein.

Comparing Figures 5.2 and 5.3, we see that two differences exist between the RF and baseband transmitter models. Firstly the frequency conversion and amplification stages are swapped, which is to be expected since we have gone from an RF to a baseband model. Secondly and most importantly, the amplifier system operator changes from the pure Volterra Series to the Baseband Volterra Series. For comparison purposes, we repeat both of these operators below in terms of the general signal $x(t)$:

RF Amplifier Operator:

$$\hat{\boldsymbol{A}}[x(t)] = \sum_{n=1}^{\infty} \hat{\boldsymbol{A}}_n[x(t)] \tag{5.29}$$

where

$$\hat{\boldsymbol{A}}_n[x(t)] = \int_0^{\infty} \cdots \int_0^{\infty} \hat{a}_n(\tau_1, \ldots, \tau_n)\, x(t-\tau_1) \cdots x(t-\tau_n)\, d\tau_1 \cdots d\tau_n \tag{5.30}$$

Baseband Amplifier Operator:

$$\boldsymbol{A}\big[x(t)\big] = \sum_{\text{odd } n=1}^{\infty} \boldsymbol{A}_n\big[x(t)\big] \tag{5.31}$$

where

$$\boldsymbol{A}_n\big[x(t)\big] = \int_0^{\infty} \cdots \int_0^{\infty} a_n(\tau_1, \ldots, \tau_n)\, x(t-\tau_1) \cdots x(t-\tau_{\lceil \frac{n}{2} \rceil})\, x^*(t-\tau_{\lceil \frac{n}{2} \rceil+1}) \cdots x^*(t-\tau_n)\, d\tau_1 \cdots d\tau_n \tag{5.32}$$

Since we are not intending to use a *Model Based Derivation* strategy for predistortion filter parameter estimation, the RF to baseband kernel conversion $\hat{a}_n(\tau_1, \ldots, \tau_n)$ to $a_n(\tau_1, \ldots, \tau_n)$ given by (5.24) is irrelevant in our predistortion work, though it must be pointed out that this conversion leaves the baseband kernel being complex in general. What is relevant in our work however is the removal of even order op-

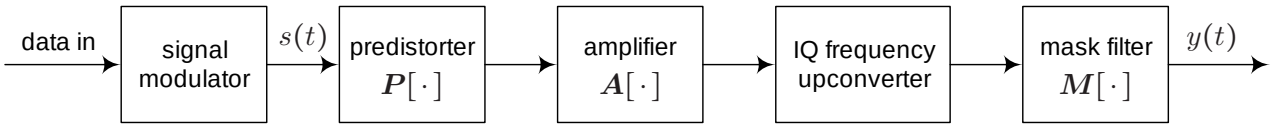


Figure 5.4: Baseband transmitter model with predistortion filter inserted

erators, compare (5.29) and (5.31), and also the introduction of conjugated product terms, compare (5.30) and (5.32), because both have a direct bearing on the final architecture of the predistortion filter.

5.3 Predistortion Filter Architecture

As discussed in Chapter 1, the digital predistortion process involves inserting a nonlinear digital filter directly at the output of the signal modulator. This filter's transfer characteristic is designed to be the inverse of the power amplifier's, thereby creating an overall linear transmission path. Insertion of the digital predistortion filter into the baseband transmitter model is shown in Figure 5.4.

It would now be logical to assume that the predistortion filter architecture with the greatest potential for realizing this inverted transfer characteristic would be one that shares the same general architecture as the cascaded amplifier operator $\boldsymbol{A}[\cdot]$, thus capturing similar dynamic memory effects, while exhibiting a somewhat inverted set of kernel coefficients, thus performing the complementary dynamic compensation. Based on this general thinking, we let the predistortion filter in our work take on the same architecture as (5.31) and (5.32) but with a temporal discretization since filtering is to be implemented digitally. An operator name change to $\boldsymbol{P}[\cdot]$ along with a nonlinear order re-indexing from n to m is also required as presented below:

Predistortion Filter Operator:

$$\boldsymbol{P}\big[\, s[k] \,\big] = \sum_{\text{odd } m=1}^{\infty} \boldsymbol{P}_m\big[\, s[k] \,\big] \tag{5.33}$$

where

$$\boldsymbol{P}_m\big[\, s[k] \,\big] = \sum_{i_1=0}^{\infty} \cdots \sum_{i_m=0}^{\infty} p_m[i_1, \ldots, i_m]\, s[k-i_1] \cdots s[k-i_{\lceil \frac{m}{2} \rceil}]\, s^*[k-i_{\lceil \frac{m}{2} \rceil+1}] \cdots s^*[k-i_m] \tag{5.34}$$

Here, k and i represent the discrete-time and delay variables respectively and $p_m[i_1, \ldots, i_m]$ represents the predistortion filter's m^{th} order Baseband Volterra kernel. In Chapter 12,

we address the need to prune this predistortion filter kernel. Prior to this however, we work in terms of the unpruned (5.33) and (5.34) in order to maximize insight into algorithm development.

The important point to take away from this chapter is that whilst we judiciously choose to model the RF power amplifier by the pure Volterra Series, the actual predistortion filter ends up being modeled by the Baseband Volterra Series variant. This result is attributed to the changing form of the power amplifier operator during the RF to baseband transmitter model conversion.

Chapter 6

Digital Predistortion In The Time-Domain

In this chapter we investigate the time-domain nonlinear interactions between predistorter and amplifier with the goal of understanding how the predistorter is able to cancel the nonlinear effects of the power amplifier and hence achieve linearization.

6.1 Intuitive Graphical Analysis

We begin with an intuitive graphical analysis of how the predistortion concept works in the time-domain. Consider Figure 6.1 which presents the AM-AM characteristic of a *quasi-memoryless* power amplifier[1]. Here it can be seen that as input signal amplitude increases, output signal amplitude starts to compress before eventually saturating, leaving the amplifier with a non-constant gain and a precise linear working region.

Without predistortion, the amplifier's input signal is scaled such that its amplitude generally remains within the linear region as depicted by the left most Probability Density Function (PDF) at the bottom of the figure. In this case, distortion is avoided but the amplifier is running inefficiently with severe output back off[2].

Consider now the scenario of predistortion. The predistorter's input signal is scaled such that its peak amplitude equals C_I as depicted by the middle PDF at the bottom of the figure. For those upper amplitudes now within the amplifier's nonlinear region (A_I to C_I), the predistorter performs expansion in order to compensate for the amplifier's impending compression. For example, the predistorter

[1]The same general concepts about to be covered extend to the dynamic case but are easier to comprehend without AM-AM hysteresis.

[2] [252] states that Crest Factor is commonly used to determine the amplifier's output back off but believes this is not always the correct method since it is the statistical envelope distribution that will determine average distortion. We agree with this line of thinking.

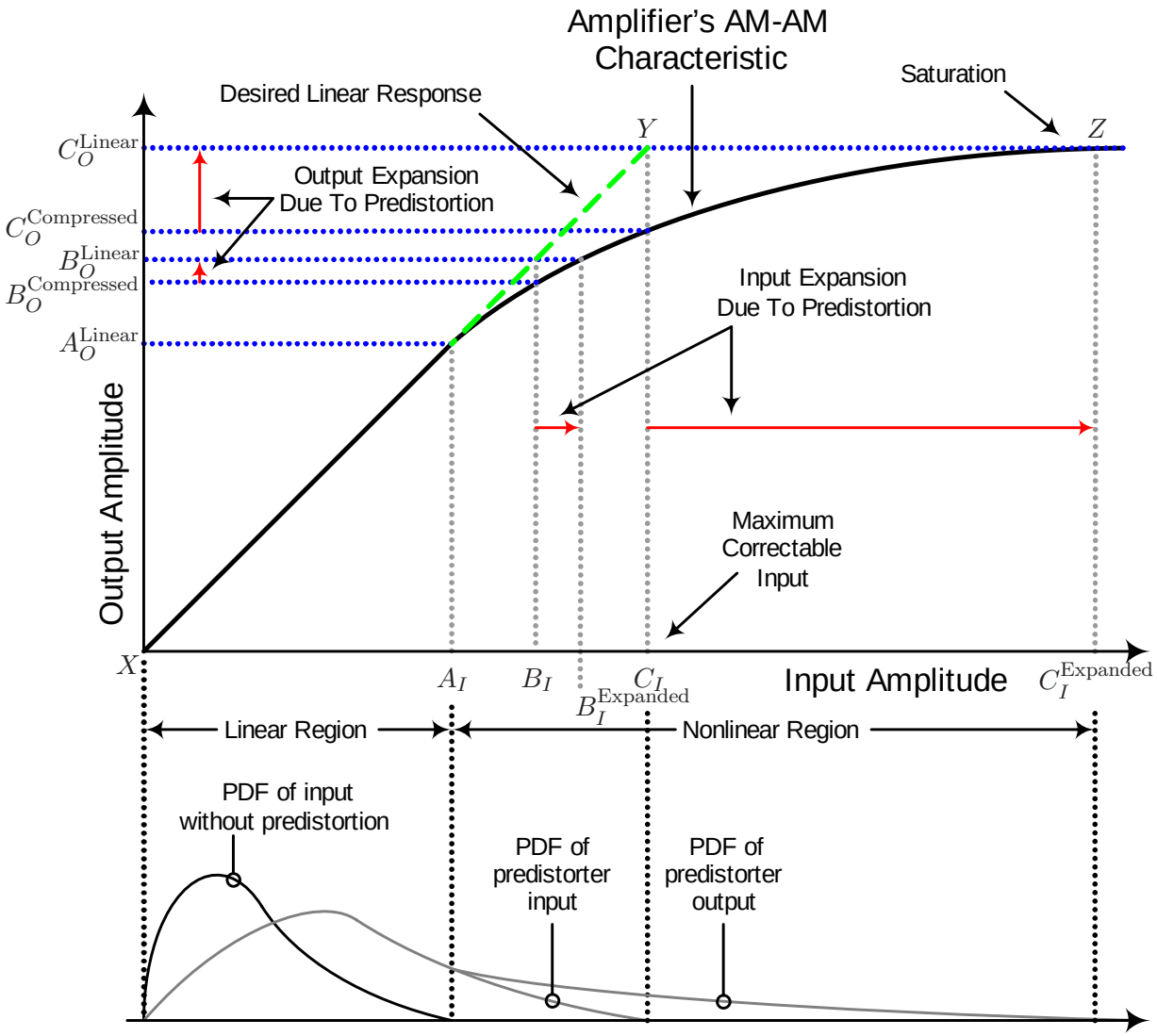


Figure 6.1: Graphical analysis of digital predistortion in the time-domain

will expand input B_I to B_I^{Expanded} so that the amplifier output also expands from $B_O^{\text{Compressed}}$ to the desired B_O^{Linear}. In this way, the amplifier is able to operate *efficiently* within its nonlinear region yet still appear linear. Predistortion does have its practical limits however with C_I representing the maximum correctable input and hence upper limit to linearization. Beyond this point, signal expansion by the predistorter is capped by full output saturation. Overall, this gives the predistorted power amplifier a maximally hard saturation characteristic represented by line segments X-Y-Z of Figure 6.1 [131, 153].

It should not be forgotten that AM-PM related distortion is also corrected for during the predistortion process, however it is this signal expansion concept which most intuitively represents the workings of the digital predistortion filter.

6.2 Mathematical Operator Analysis

We now turn attention to a mathematical operator analysis of the interactions between predistorter and power amplifier. In the previous chapter, we derived the baseband equivalent transmitter model and inserted the predistortion filter at the output of the signal modulator according to Figure 5.4, page 52. The cascaded predistorter $\boldsymbol{P}[\,\cdot\,]$ and power amplifier $\boldsymbol{A}[\,\cdot\,]$ were both represented by the Baseband Volterra Series, now assumed to be discretized. Figure 6.2 presents this cascade along with a unique representation of its output which we have pioneered and refer to as the *Distortion Array.*

Effectively, the *Distortion Array* is a graphical organizing tool for keeping track of those nonlinear distortion components generated by the cascade. Its elements are either blank or ticked. A blank element signifies nil distortion whilst a ticked element signifies one or more distortion components. Ticks are also color coded to classify the origin of components. Each row of the array is associated with components generated by the corresponding in-feeding amplifier operator whilst each column of the array is associated with components of equivalent order as labeled. The output of the n^{th} order amplifier operator (n odd) is given by:

$$\boldsymbol{A}_n\Big[\,\boldsymbol{P}\big[\,s[k]\,\big]\,\Big] = \boldsymbol{A}_n\Bigg[\sum_{\text{odd } m=1}^{\infty} \boldsymbol{P}_m\big[\,s[k]\,\big]\Bigg] \tag{6.1}$$

Expanding the right hand side of (6.1) in terms of n-linear operators then gives:

$$\boldsymbol{A}_n\Big[\,\boldsymbol{P}\big[\,s[k]\,\big]\,\Big] = \sum_{\text{odd } m_1=1}^{\infty} \cdots \sum_{\text{odd } m_n=1}^{\infty} \boldsymbol{A}_n\Big\{\,\boldsymbol{P}_{m_1}\big[\,s[k]\,\big],\ldots,\boldsymbol{P}_{m_n}\big[\,s[k]\,\big]\,\Big\} \tag{6.2}$$

Each n-linear operator $\boldsymbol{A}_n\big\{\,\boldsymbol{P}_{m_1}\big[\,s[k]\,\big],\ldots,\boldsymbol{P}_{m_n}\big[\,s[k]\,\big]\,\big\}$ in (6.2) represents a single distortion component, which by definition (5.6), is of nonlinear order $(m_1+\cdots+m_n)$. Each of these components make up the *Distortion Array* row being fed by amplifier operator $\boldsymbol{A}_n[\,\cdot\,]$. Since both n and m are odd, each component will be of odd nonlinear order. From (6.2), we can make some very important observations:

1. The minimum order of distortion generated by $\boldsymbol{A}_n[\,\cdot\,]$ is n, corresponding to component $\boldsymbol{A}_n\big\{\,\boldsymbol{P}_1\big[\,s[k]\,\big],\ldots,\boldsymbol{P}_1\big[\,s[k]\,\big]\,\big\}$. In terms of the *Distortion Array*, this means that the row being fed by $\boldsymbol{A}_n[\,\cdot\,]$ commences at the n^{th} order column, and a leading diagonal forms across the array. Furthermore, since digital predistortion does not attempt to compensate for an amplifier's linear distortion, $\boldsymbol{P}_1[\,\cdot\,]$ can be assumed transparent, that is its kernel is the unit impulse, and $\boldsymbol{P}_1\big[\,s[k]\,\big] = s[k]$. In effect, these minimum order leading diagonal components of the array then represent pure amplifier distortion

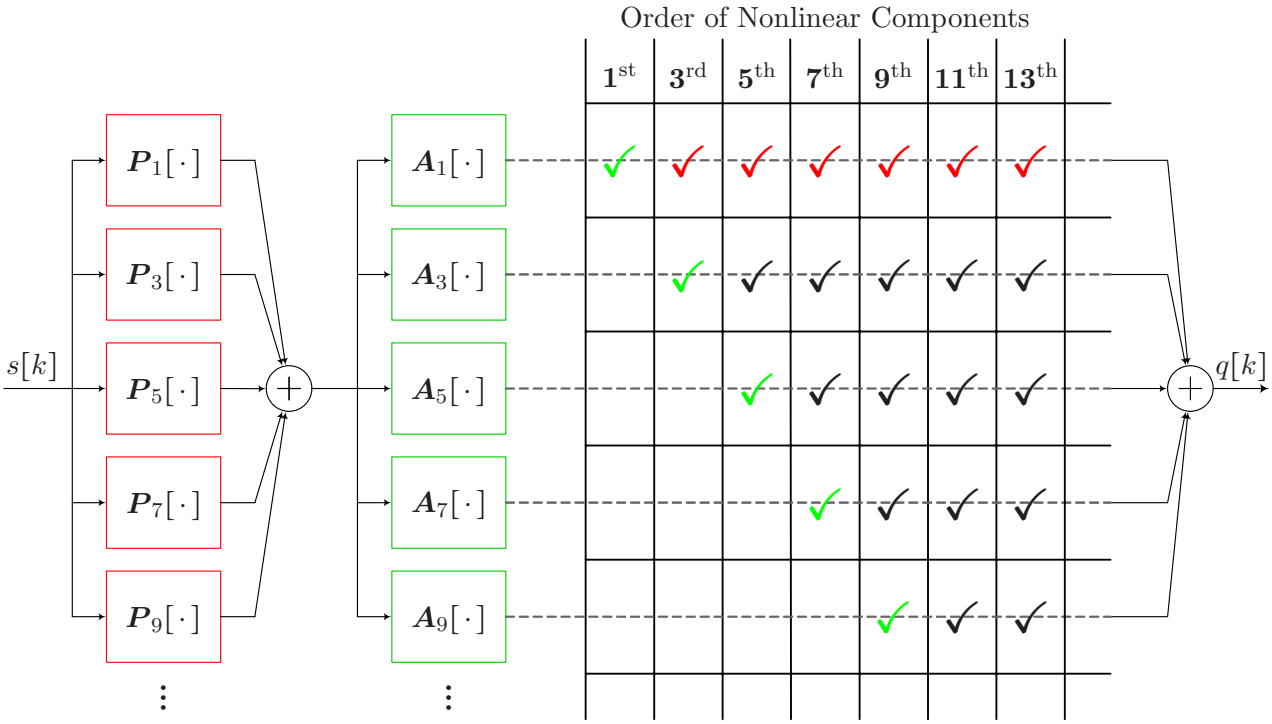


Figure 6.2: Predistorter-Amplifier cascade (left) and *Distortion Array* (right)

$\boldsymbol{A}_n\{\, s[k], \ldots, s[k] \,\} = \boldsymbol{A}_n[\, s[k] \,]$. To reflect this, the leading diagonal ticks are colored green, the same color as the amplifier operator blocks. These components need to be canceled as per the original linearization problem.

2. Since $\boldsymbol{A}_1[\,\cdot\,]$ is a linear operator, its output components are merely linearly filtered predistorter components. For the purposes of this analysis, we can assume negligible linear amplifier distortion and as such approximate the output of $\boldsymbol{A}_1[\,\cdot\,]$ to be the predistorter components scaled by the gain G of the amplifier. Because of this relationship with the predistorter, ticks along the first row of the *Distortion Array* are colored red, the same color as the predistorter operator blocks. As will become evident shortly, since these components avoid nonlinear amplifier inter-modulation, they will be used as the tools for canceling other *Distortion Array* components of equivalent order.

3. We refer to the elements of the *Distortion Array* that aren't on the leading diagonal (green) or in the first row (red) as *parasitic* elements since they represent unwanted by-products of inserting the predistortion filter, specifically inter-modulation components $\boldsymbol{A}_n\{\, \boldsymbol{P}_{m_1}[\, s[k] \,], \ldots, \boldsymbol{P}_{m_n}[\, s[k] \,] \,\}$ where $n \neq 1$ and $(m_1, \cdots, m_n)$ don't all equal 1. These components are neither pure amplifier nor predistorter. *Parasitic* element ticks are colored black in the *Distortion Array* to signify their unwanted nature.

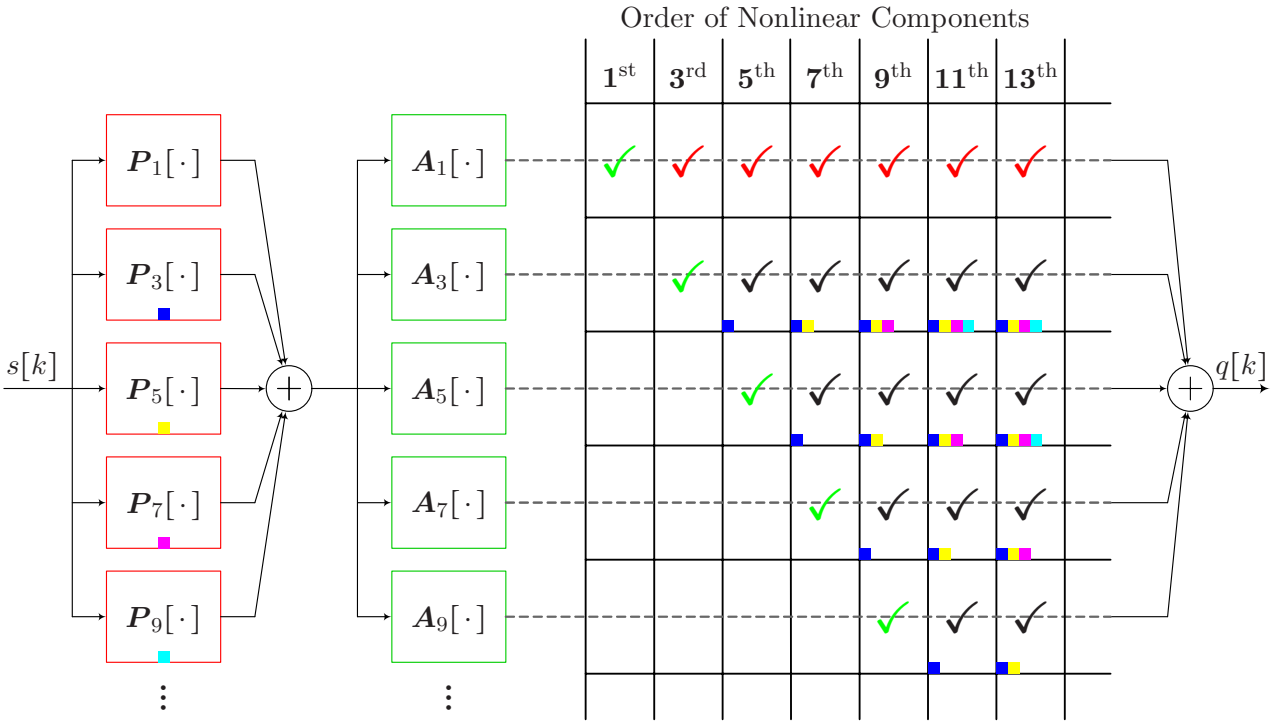


Figure 6.3: *Parasitic* components generated by $\boldsymbol{P}_m[\,\cdot\,]$

Parasitic components generated by $\boldsymbol{P}_m[\,\cdot\,]$ ($m \geq 3$) will be of nonlinear order greater than m. In the *Distortion Array*, these *parasitic* components will be bounded to the left by a diagonal line ***1)*** running parallel to the green leading diagonal and ***2)*** passing through the m^{th} order element of the first row. Figure 6.3 demonstrates this by associating each *parasitic* element of the *Distortion Array* with its generating predistorter operator block by way of a small colored box. Those *parasitic* elements which possess more than one small colored box subsequently represent multiple *parasitic* components of different origin. For example the two components:

$$\boldsymbol{A}_3\Big\{\, \boldsymbol{P}_1\big[\, s[k]\,\big],\, \boldsymbol{P}_3\big[\, s[k]\,\big],\, \boldsymbol{P}_3\big[\, s[k]\,\big] \,\Big\} \tag{6.3}$$

$$\boldsymbol{A}_3\Big\{\, \boldsymbol{P}_1\big[\, s[k]\,\big],\, \boldsymbol{P}_1\big[\, s[k]\,\big],\, \boldsymbol{P}_5\big[\, s[k]\,\big] \,\Big\} \tag{6.4}$$

are both 7^{th} order *parasitic* components represented by the single element in the 7^{th} order column of the row being fed by $\boldsymbol{A}_3[\,\cdot\,]$. (6.3) is generated by $\boldsymbol{P}_3[\,\cdot\,]$ (small blue box) while (6.4) is generated by $\boldsymbol{P}_5[\,\cdot\,]$ (small yellow box).

6.3 Ideal Predistorter Operator

To take the next step in understanding how each predistorter operator is configured to linearize the power amplifier, we must view the predistorter-amplifier cascade in terms of its equivalent cascade nonlinearity $\boldsymbol{Q}[\,\cdot\,]$. Figure 6.4 presents this equivalent cascade nonlinearity in terms of its individual operators $\boldsymbol{Q}_l[\,\cdot\,]$ along with the original representation of the predistorter-amplifier cascade and *Distortion Array*. We immediately see that $\boldsymbol{Q}_l\big[\,s[k]\,\big]$ is the sum of components represented by the l^{th} order column of the *Distortion Array*:

$$\boldsymbol{Q}_1\big[\,s[k]\,\big] = G\,s[k] \tag{6.5}$$

$$\boldsymbol{Q}_3\big[\,s[k]\,\big] = \boldsymbol{A}_3\big[\,s[k]\,\big] + G\,\boldsymbol{P}_3\big[\,s[k]\,\big] \tag{6.6}$$

$$\text{For odd } l \geq 5 \quad \boldsymbol{Q}_l\big[\,s[k]\,\big] = \boldsymbol{A}_l\big[\,s[k]\,\big] + G\,\boldsymbol{P}_l\big[\,s[k]\,\big] + \boldsymbol{U}_l\big[\,s[k]\,\big] \tag{6.7}$$

In each of (6.5), (6.6) and (6.7), terms are color coded in accordance with the elements of the *Distortion Array* to which they correspond; namely amplifier (green), predistorter (red) and parasitic (black). In (6.7), $\boldsymbol{U}_l\big[\,s[k]\,\big]$ represents the set of $\boldsymbol{U}$nwanted l^{th} order parasitic components:

$$\boldsymbol{U}_l\big[\,s[k]\,\big] = \sum_{\text{odd } n=3}^{l-2} \left[\sum_{\boldsymbol{m}_{n_l}} \boldsymbol{A}_n\Big\{ \boldsymbol{P}_{m_1}\big[\,s[k]\,\big], \cdots, \boldsymbol{P}_{m_n}\big[\,s[k]\,\big] \Big\} \right] \tag{6.8}$$

Here, $\boldsymbol{m}_{n_l}$ is the vector subspace of $\big\{\,[m_1, \cdots, m_n]\,\big\}$ for which $(m_1 + \cdots + m_n = l)$. Consistent with our earlier findings, we see from (6.8) that l^{th} order parasitic components are generated by lower order predistorter operators.[3]

It can now be seen from (6.6) that $\boldsymbol{Q}_3\big[\,s[k]\,\big]$ can be eliminated and hence 3^{rd} order amplifier linearization achieved if:

$$\boldsymbol{P}_3\big[\,s[k]\,\big] = \frac{-\boldsymbol{A}_3\big[\,s[k]\,\big]}{G} \tag{6.9}$$

Similarly from (6.7), for odd $l \geq 5$, $\boldsymbol{Q}_l\big[\,s[k]\,\big]$ can be eliminated and hence l^{th} order amplifier linearization achieved if:

$$\boldsymbol{P}_l\big[\,s[k]\,\big] = \frac{-\Big(\boldsymbol{A}_l\big[\,s[k]\,\big] + \boldsymbol{U}_l\big[\,s[k]\,\big]\Big)}{G} \tag{6.10}$$

[3]For clarification, since each n-linear operator $\boldsymbol{A}_n\{\cdots\}$ of (6.8) can be represented as an accumulation of standard operators $\boldsymbol{A}_n[\,\cdot\,]$ acting on partial input accumulations [240, 248], $\boldsymbol{U}_l\big[\,\cdot\,\big]$ can be considered a Baseband Volterra operator in its most fundamental form.

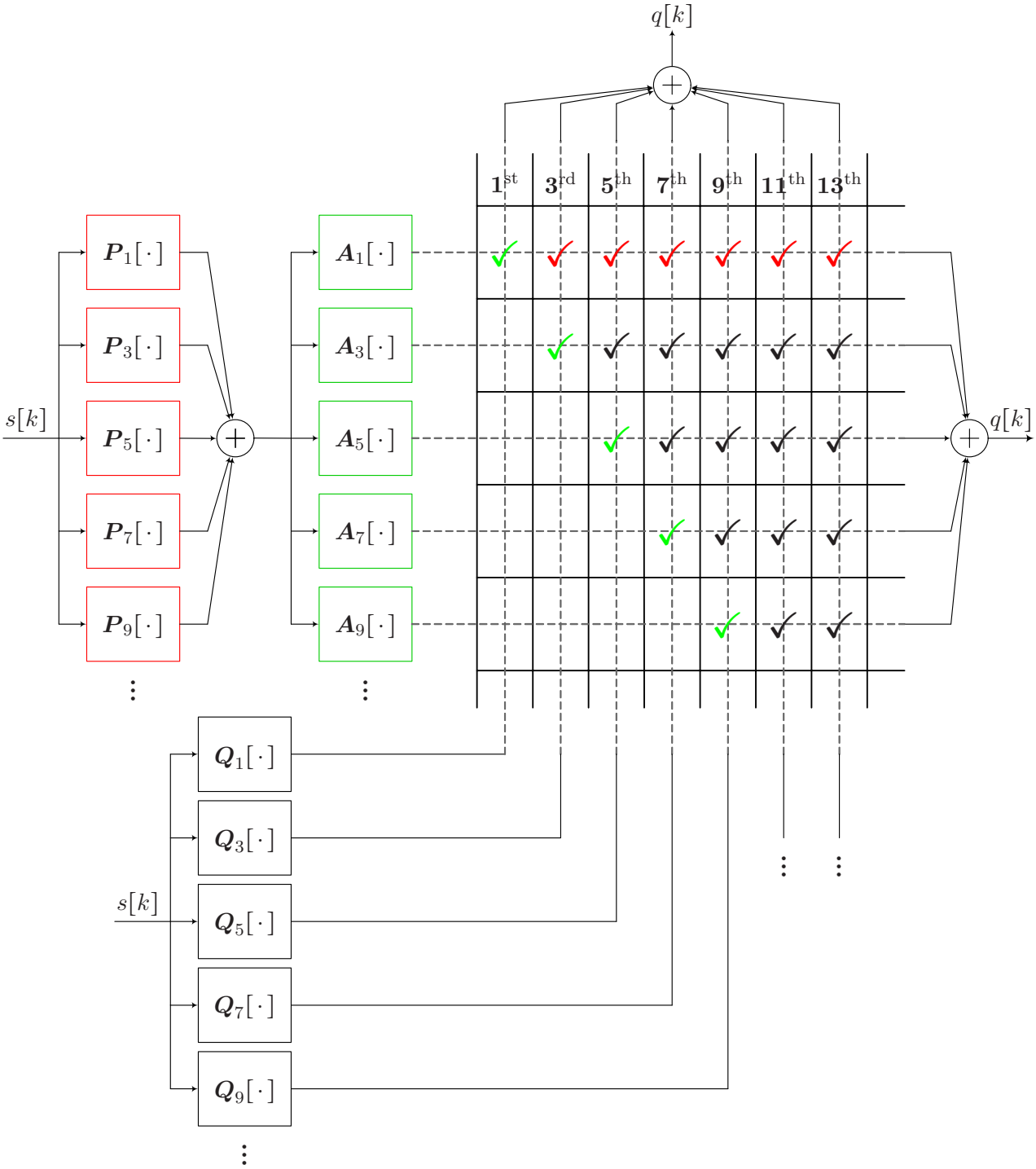


Figure 6.4: Equivalent cascade nonlinearity $\boldsymbol{Q}[\cdot]$

So for odd $l \geq 5$, not only is $\boldsymbol{P}_l\big[\,\cdot\,\big]$ required to cancel $\boldsymbol{A}_l\big[\,\cdot\,\big]$ (the original l^{th} order amplifier nonlinearity), but it's also required to cancel the l^{th} order parasitic components generated by lower order predistorter operators.

The implication of this last fact is that whilst any single order of amplifier linearization can be achieved according to (6.9) and (6.10), such a linearization will become obsolete if a lower order predistorter operator changes for any reason. It follows that if one plans to linearize the amplifier up to l^{th} order, computation of predistorter operators must occur in ascending order up to l^{th} order.

6.4 Theoretical Limits of Digital Predistortion

Despite the fact that any single order of amplifier linearization is achievable according to (6.9) and (6.10), entire linearization is not theoretically possible. This is quite simply because every time a predistorter operator is computed to eliminate amplifier and / or parasitic distortion of equivalent order, it produces higher order parasitic components in the process. So no matter how high an order one wishes to linearize up to, at least parasitic distortion will remain.

Thankfully, we can assume higher order parasitic components are ***1)*** generally uncorrelated and therefore don't add constructively and ***2)*** are of progressively lower power. This means that despite not ever being able to theoretically achieve entire linearization, parasitic distortion can be reduced with each higher order of linearization. In practice, predistortion is performed for 3^{rd} order up to some finite maximum order. Choosing this maximum order is a trade off between amplifier / parasitic distortion levels (keeping in mind regulatory spectral mask and terminal sensitivity requirements) and predistorter computational complexity. This trade off will be discussed in more detail in Chapter 8.

Before leaving this chapter, it is worth noting that this operator analysis of digital predistortion has strong links with the P^{th} Order Inverses technique, a Model Based Derivation strategy first introduced within the literature review. The Distortion Array was in fact developed here as a means of more intuitively portraying the central concepts of this technique which are often overshadowed by mathematical rigor. This mathematical rigor is a result of expanding the $\boldsymbol{U}_l\big[\,\cdot\,\big]$ operator of (6.8) *in terms of its standard Baseband Volterra operators. The aim of this expansion is to derive the exact form of the linearizing predistorter operators* (6.10)*. Such an exact form is unnecessary in our work however since we intend using a Self-Learning strategy rather than Model Based Derivation to compute predistorter kernels. In other words, in our case, concept is more important than precise mathematical form. Discussion of the intended Self-Learning strategy begins in Chapter 9.*

Chapter 7

Digital Predistortion In The Frequency-Domain

In the previous chapter, we analyzed the time-domain nonlinear interactions existing between predistorter and amplifier and were able to gain an intuitive understanding of how the predistorter achieves linearization. In this chapter we extend this insight to the frequency-domain in order to show how the amplifier output spectrum behaves during the linearization process.

Predistortion in the frequency-domain is best viewed in terms of the equivalent cascade nonlinearity $\boldsymbol{Q}[\,\cdot\,]$, introduced in the previous chapter and repeated in Figure 7.1. Here, individual operators $\boldsymbol{Q}_l[\,\cdot\,]$ are represented in terms of their constituent amplifier, predistorter and parasitic distortion operators according to (6.5), (6.6) and (6.7). It is worth noting that since both predistorter and amplifier are modeled as Baseband Volterra systems, the equivalent cascade nonlinearity $\boldsymbol{Q}[\,\cdot\,]$ can also be modeled as such.

Consider now the general l^{th} order cascade nonlinearity $\boldsymbol{Q}_l[\,\cdot\,]$ with input random process $s[k]$. Let $s[k]$ be representative of our OFDM / CDMA target signal modulations, that is, complex baseband with continuous-time bandwidth B centered at 0 Hz. This is depicted by the power spectral density $\mathcal{S}_{ss}(f)$ in the bottom left of Figure 7.1. A defining characteristic of nonlinear systems is spectral regrowth. Since l-fold time-domain multiplication within $\boldsymbol{Q}_l[\,\cdot\,]$'s representative Baseband Volterra Series is equivalent to l-fold frequency-domain convolution [307], $\boldsymbol{Q}_l[\,\cdot\,]$ will generate a complex baseband output process $q_l[k]$ with continuous-time bandwidth lB centered at 0 Hz. This is depicted by the power spectral density $\mathcal{S}_{q_l q_l}(f)$ in the bottom right of Figure 7.1.

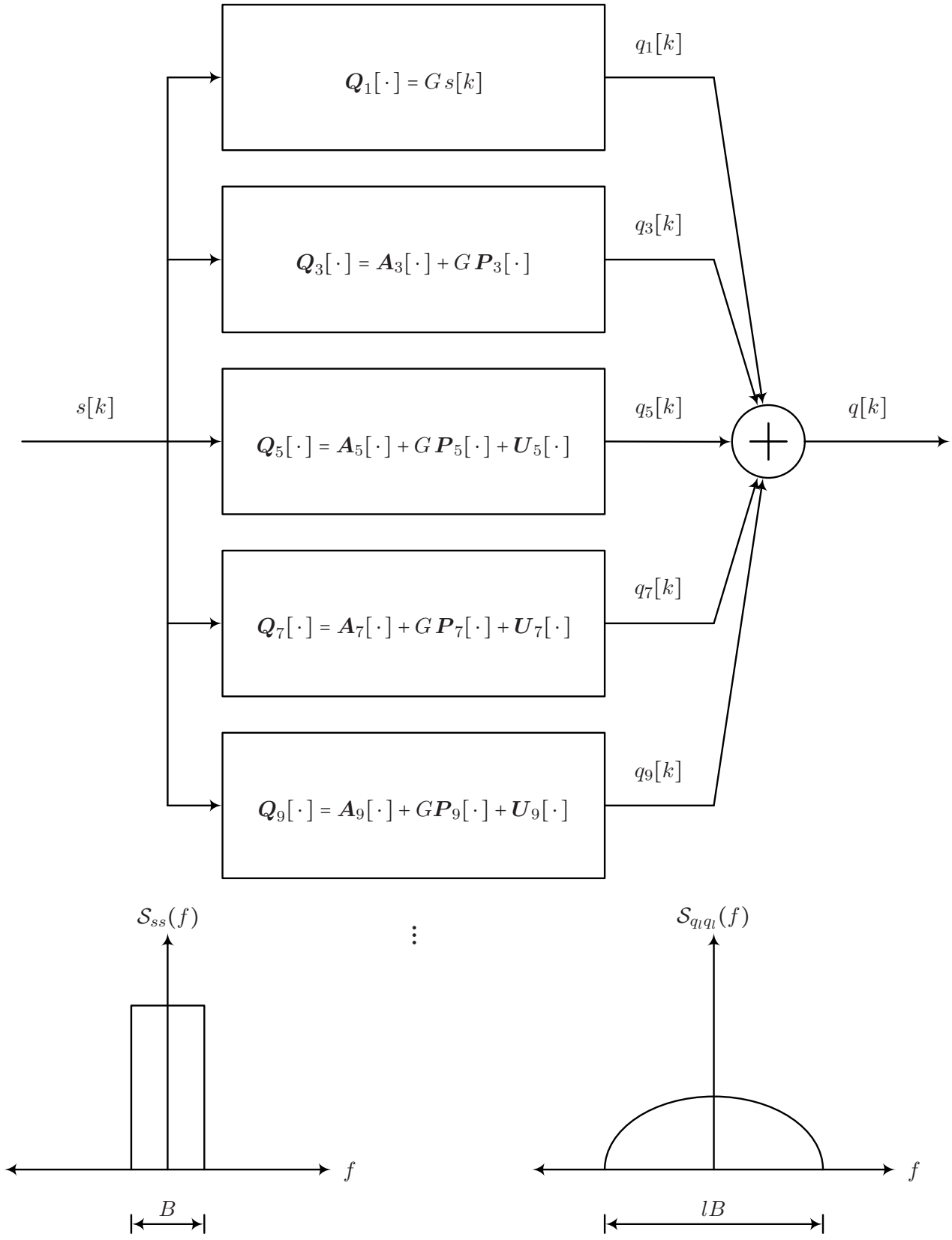


Figure 7.1: Equivalent cascade nonlinearity $\boldsymbol{Q}[\,\cdot\,]$

While l-fold spectral regrowth and a 0 Hz center frequency is guaranteed, the level of $\mathcal{S}_{q_l q_l}(f)$ will ultimately depend on the makeup of the nonlinearity $\boldsymbol{Q}_l[\cdot]$ and therefore the current state of the predistorter. Since practical predistortion is generally performed for 3rd order up to some finite maximum order (refer to Section 6.4), we will examine the levels of $\mathcal{S}_{q_l q_l}(f)$ at each progressively higher predistortion state and draw conclusions.

First consider $\mathcal{S}_{q_l q_l}(f)$ prior to predistortion. In this state, $\boldsymbol{Q}_l[\cdot] = \boldsymbol{A}_l[\cdot]$ and output distortion is pure amplifier distortion. This is shown in Figure 7.2 for a real 25 Watt Class-AB push-pull power amplifier with DAB signal modulation ($B = 1.537$ MHz). It is noted that higher orders of distortion are present within this figure but their spectral envelopes are hidden beneath the noise floor.

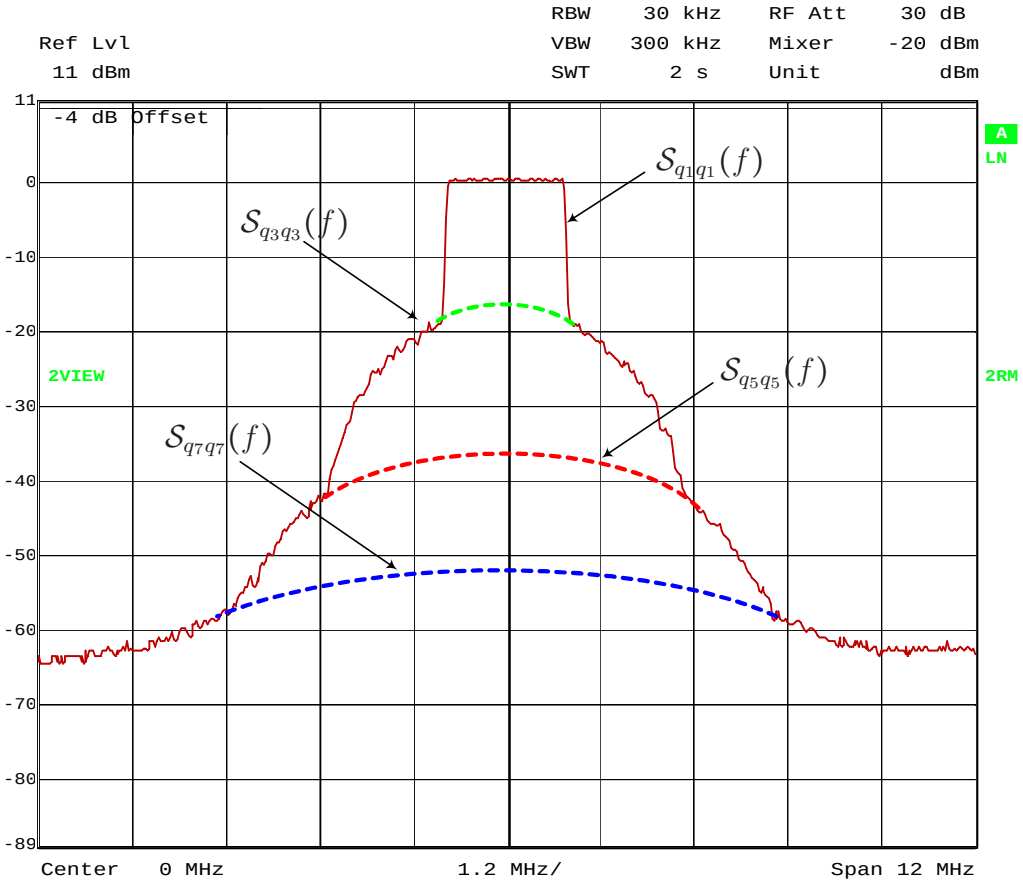


Figure 7.2: Power spectra $\mathcal{S}_{q_l q_l}(f)$ prior to predistortion for a real 25 Watt Class-AB push-pull power amplifier with DAB signal modulation

Now assume 3rd order predistortion is performed and consider the level of $\mathcal{S}_{q_l q_l}(f)$. Recall from Chapter 6 and the *Distortion Array* that 3rd order predistortion eliminates $\boldsymbol{Q}_3[\,\cdot\,]$, and hence $\mathcal{S}_{q_3 q_3}(f)$, but in the process generates higher order parasitic distortion. It follows that $\boldsymbol{Q}_l[\,\cdot\,]$ for $l \geq 5$ will change from pure amplifier distortion to $\boldsymbol{Q}_l[\,\cdot\,] = \boldsymbol{A}_l[\,\cdot\,] + \boldsymbol{U}_l[\,\cdot\,]$ and $\mathcal{S}_{q_l q_l}(f)$ will experience a slight growth. This spectral behavior is illustrated in Figure 7.3.

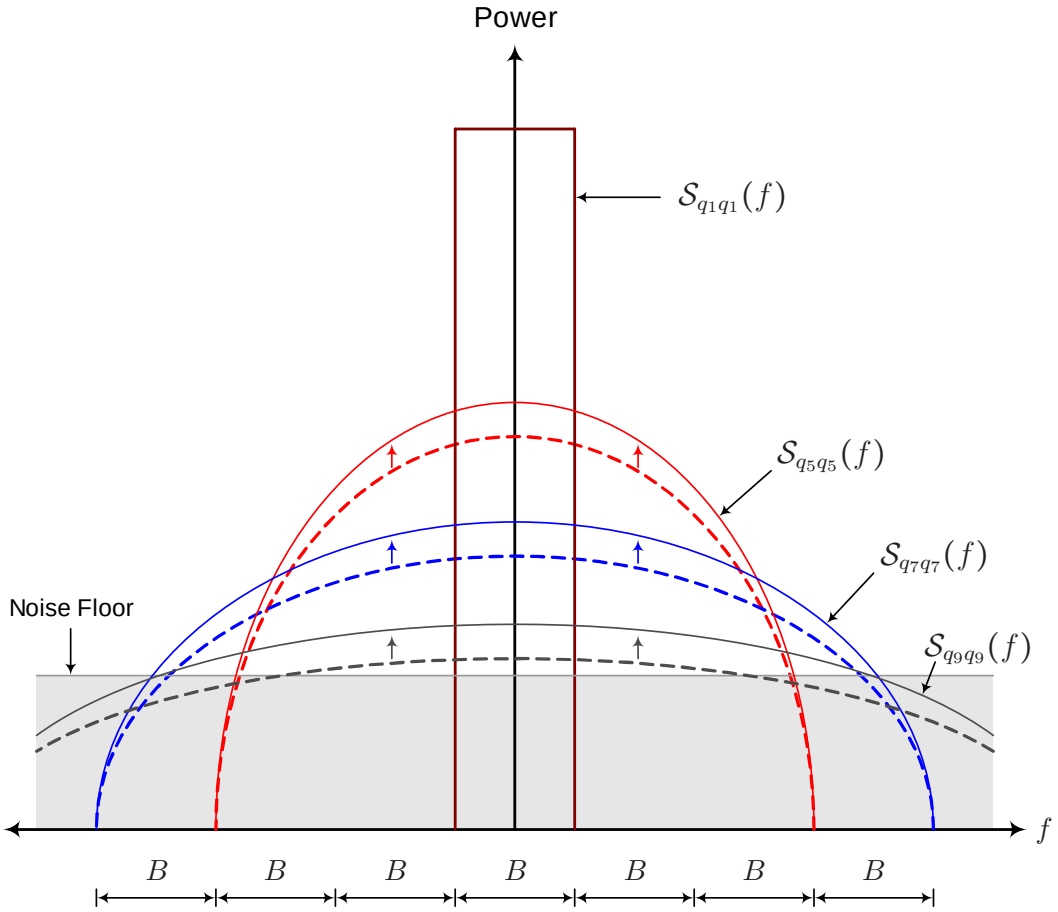


Figure 7.3: Power spectra $\mathcal{S}_{q_l q_l}(f)$ after 3rd order predistortion (solid traces). Dashed traces represent levels prior to predistortion and arrows highlight parasitic growth. 3rd order distortion is totally eliminated. Higher orders of distortion are present beneath the noise floor but aren't shown to avoid clutter.

A real example of 3rd order predistortion is also presented in Figure 7.4 for a 25 Watt Class-AB push-pull power amplifier with DAB signal modulation. It can be seen here that after 3rd order predistortion, the distortion characteristic is less rounded; indicating that only higher order distortion remains. Also, this higher order distortion becomes visible above the noise floor where it wasn't visible before; a clear demonstration of parasitic growth. It must be noted that Figure 7.4 exhibits a 10 dB higher noise floor compared to Figure 7.2, despite both being associated with the same amplifier and signal modulation. This is due to the finite resolution of reconstruction DACs and the heavier scaling of their input signals to accommodate the impending Crest Factor growth of predistortion.

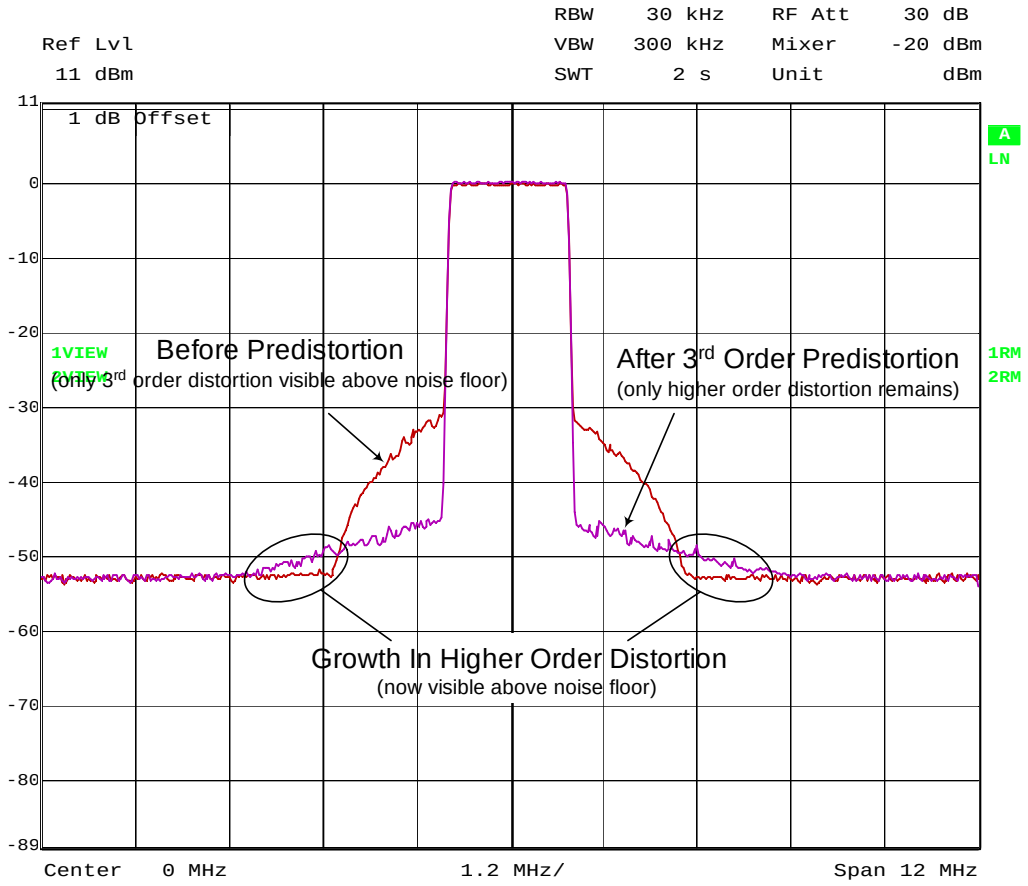


Figure 7.4: Power spectra before and after 3rd order predistortion for a real 25 Watt Class-AB push-pull power amplifier with DAB signal modulation

Now assume 3rd order predistortion is immediately followed by 5th order predistortion and consider the level of $\mathcal{S}_{q_l q_l}(f)$. Recall from Chapter 6 and the *Distortion Array* that 5th order predistortion:

1. eliminates $\boldsymbol{Q}_5[\,\cdot\,]$ and hence $\mathcal{S}_{q_5 q_5}(f)$
2. has no effect on $\boldsymbol{Q}_l[\,\cdot\,]$ for $l < 5$ and therefore $\mathcal{S}_{q_3 q_3}(f)$ remains eliminated
3. generates higher order parasitic distortion which causes $\mathcal{S}_{q_l q_l}(f)$ to grow for $l > 5$.

This spectral behavior is illustrated in Figure 7.5 in a similar manner to Figure 7.3.

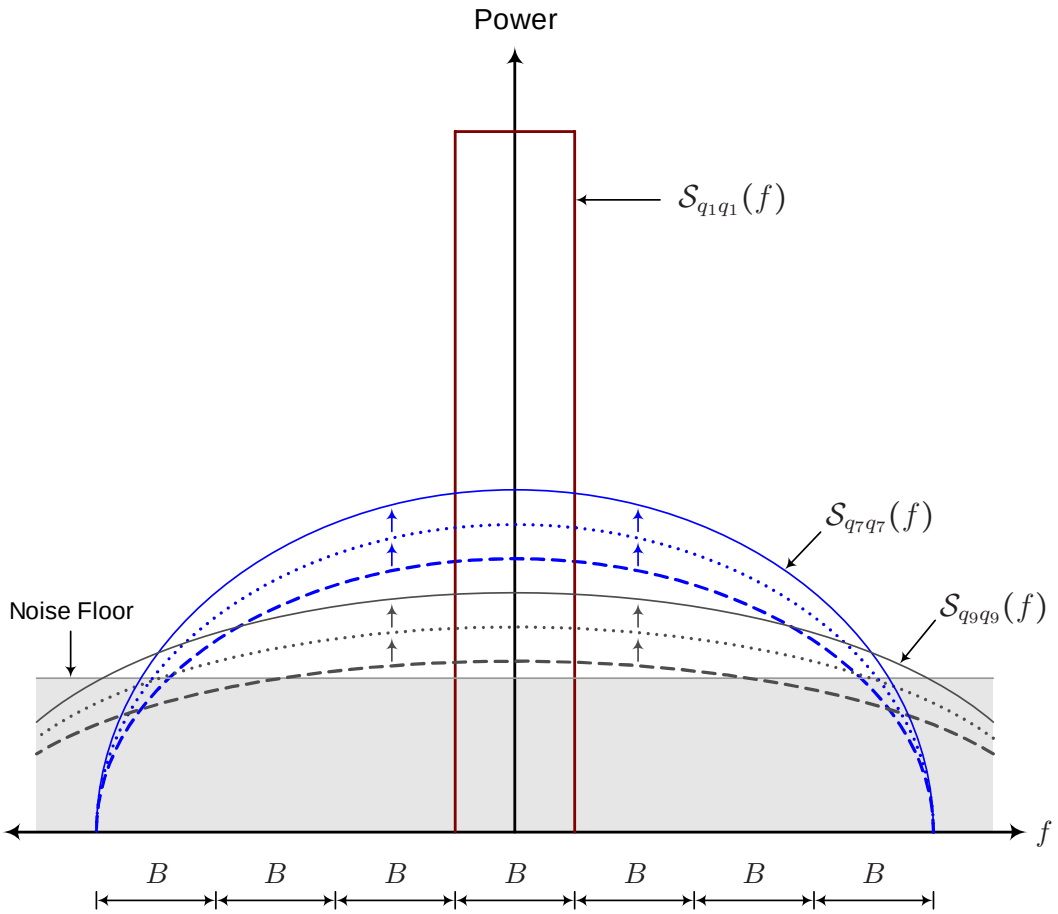


Figure 7.5: Power spectra $\mathcal{S}_{q_l q_l}(f)$ after 5th order predistortion (solid traces). Dashed traces represent levels prior to predistortion, dotted traces represent levels after 3rd order predistortion and arrows highlight parasitic growth. 3rd and 5th order distortion are totally eliminated. Higher orders of distortion are present beneath the noise floor but aren't shown to avoid clutter.

This same analysis can be continued for even higher orders of predistortion with the same conclusions drawn in each case. That is, l^{th} order predistortion will:

1. eliminate $\boldsymbol{Q}_l[\,\cdot\,]$ and therefore $\mathcal{S}_{q_l q_l}(f)$
2. have no effect on $\boldsymbol{Q}_j[\,\cdot\,]$ and therefore $\mathcal{S}_{q_j q_j}(f)$ for $j < l$
3. generate higher order parasitic distortion and therefore increase $\mathcal{S}_{q_k q_k}(f)$ for $k > l$

It is important not to be misled by Dot-Point 3 above when considering the resultant spectral distortion. While each progressive order of predistortion will generate higher order parasitics, this growth in distortion is significantly less than the reduction in distortion caused by the corresponding Dot-Point 1. Hence with each progressive order of predistortion, the resultant spectral distortion does indeed reduce.[1]

Based on the preceding analysis, the three important points to take away from this chapter are summarized below:

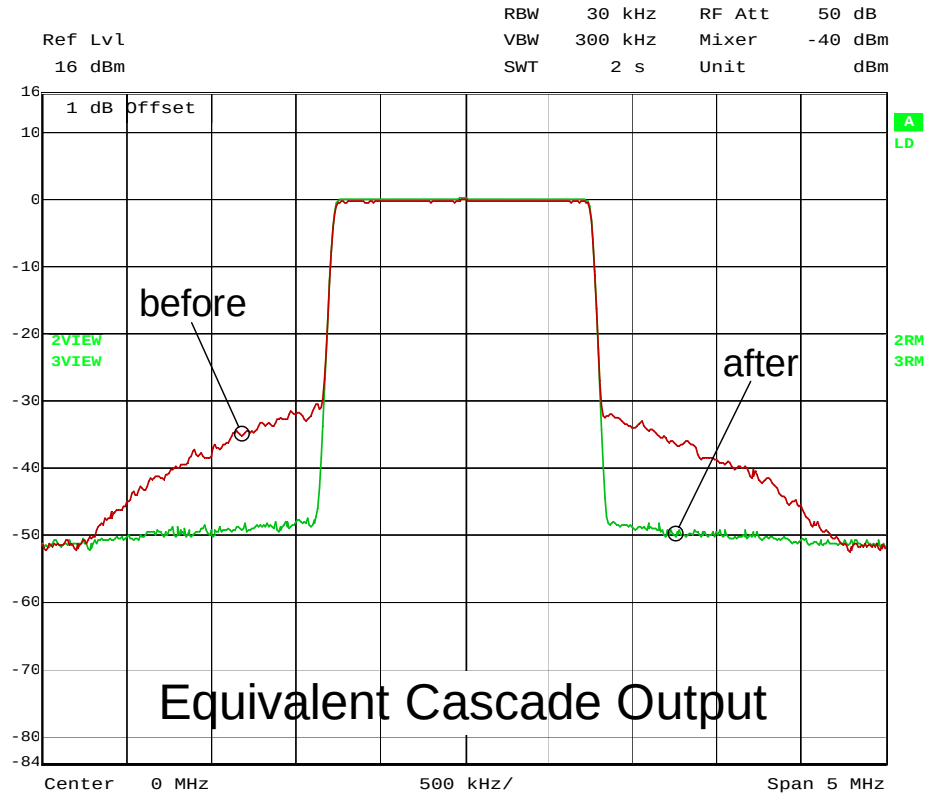
1. Equivalent cascade nonlinearity $\boldsymbol{Q}_l[\,\cdot\,]$ causes l-fold spectral regrowth.
2. Spectral regrowth from each equivalent cascade nonlinearity is collocated in frequency, specifically centered at 0 Hz.
3. With each additional order of predistortion, the resultant spectral distortion is reduced and transmitter linearization is further enhanced.

In the preceding analysis, we have been focused on the output power spectrum of the equivalent cascade nonlinearity since this is what is ultimately transmitted. Before leaving this chapter however, it is worthwhile considering the output power spectrum of the predistorter, if only to gain intuitive insight.

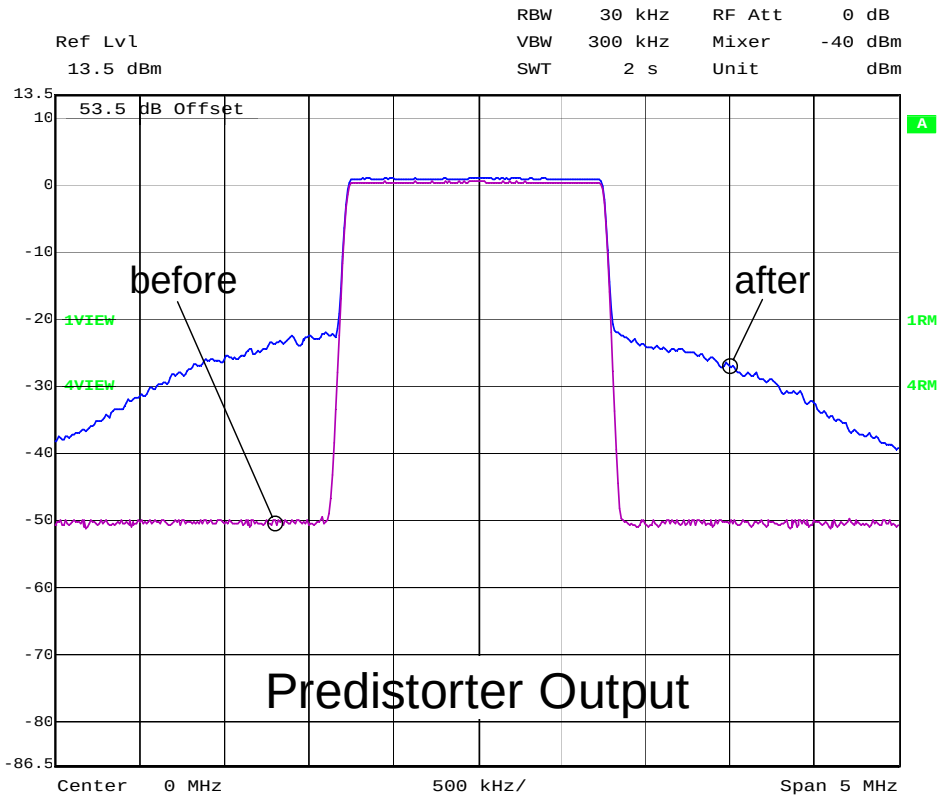
For comparison purposes, Figure 7.6a presents the output power spectrum of the equivalent cascade nonlinearity before and after a full predistortion is performed. For consistency, this is for the same 25 Watt Class-AB push-pull power amplifier and DAB signal modulation as used in earlier figures. Here we can see that predistortion successfully reduces distortion to the noise floor limit.

Figure 7.6b then presents the output power spectrum of the predistorter before and after this same full predistortion. We can see here the distortion that is created by the predistorter in order to perform the linearization. Comparing Figures 7.6a and 7.6b, we also observe that this linearizing distortion is larger than the original amplifier distortion; a clear demonstration that additional distortion is required to cancel the self-generated parasitic distortion.

[1] The level of parasitic growth illustrated in Figures 7.3 and 7.5 is indicative only.



(a) Output power spectrum of equivalent cascade nonlinearity before predistortion (red) and after predistortion (green).



(b) Output power spectrum of predistorter before predistortion (magenta) and after predistortion (blue).

Figure 7.6: Comparison of equivalent cascade and predistorter output power spectra

Chapter 8

Maximum Order Of Predistorter Nonlinearity

Now that we've covered predistortion in both the time and frequency domain, and understand the interaction between predistorter and amplifier, we are in a position to analytically choose the maximum order of predistorter nonlinearity. As already touched on in Section 6.4, this choice is a trade off between linearization performance and predistorter computational complexity. That is, the greater the maximum nonlinearity, the better the linearization performance but the higher the predistorter computational complexity. The aim is thus to find a suitable balance.

In practice, an amplifier's 3^{rd} order nonlinearity is significantly larger than its higher order nonlinearities. This was demonstrated in Figure 7.2 (page 64) for a real 25 Watt Class-AB power amplifier with DAB signal modulation. It logically follows from Chapter 6 and the *Distortion Array* that the predistorter's largest nonlinearity must also be 3^{rd} order if linearization is to be effective. This ultimately means that the parasitic distortion components generated by the large 3^{rd} order predistorter nonlinearity $\boldsymbol{P}_3[\,\cdot\,]$ inter-modulating with the large 3^{rd} order amplifier nonlinearity $\boldsymbol{A}_3[\,\cdot\,]$ will dominate all other parasitic and amplifier nonlinearities above 3^{rd} order. To be precise, these parasitic distortion components are:

$$\boldsymbol{A}_3\Big\{\, \boldsymbol{P}_3\big[\, s[k] \,\big],\, s[k],\, s[k] \,\Big\} \quad 5^{\text{th}} \text{ order} \tag{8.1}$$

$$\boldsymbol{A}_3\Big\{\, \boldsymbol{P}_3\big[\, s[k] \,\big],\, \boldsymbol{P}_3\big[\, s[k] \,\big],\, s[k] \,\Big\} \quad 7^{\text{th}} \text{ order} \tag{8.2}$$

$$\boldsymbol{A}_3\Big\{\, \boldsymbol{P}_3\big[\, s[k] \,\big],\, \boldsymbol{P}_3\big[\, s[k] \,\big],\, \boldsymbol{P}_3\big[\, s[k] \,\big] \,\Big\} \quad 9^{\text{th}} \text{ order} \tag{8.3}$$

For graphical reference, Figure 8.1 presents the *Distortion Array* with these dominant amplifier, predistorter and parasitic elements marked.

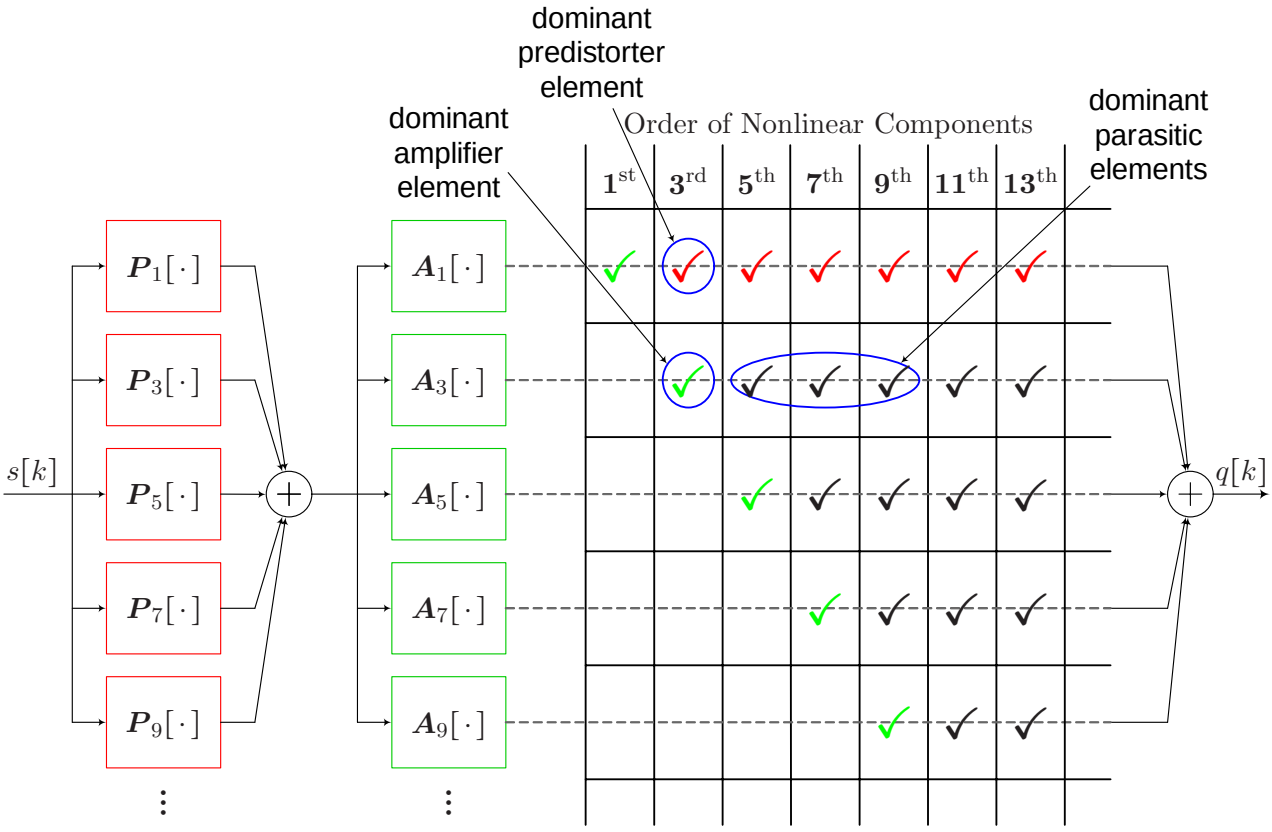


Figure 8.1: Dominant elements of the *Distortion Array*

It can be seen that if the maximum predistorter nonlinearity is chosen to be less than 9^{th} order, some of the dominant parasitic components will remain and linearization performance will be jeopardized. To avoid this, maximum predistorter order must be ≥ 9.

Considering the trade off between linearization performance and computational complexity, we favor the lower end of this range and choose not to predistort higher than 9^{th} order as this will ensure computational processing does not become a limiting factor in our work.

Chapter 9

SPFL Strategy With The New WACP Optimization Objective

As stated in the *Literature Review*, the design and analysis of a digital predistortion system is framed by two fundamental questions:

1. What architecture will the predistortion filter take?
2. How will the characterizing parameters of this filter be estimated?

In terms of Question 1, we know from Chapters 5 – 8 that the predistortion filter will be modeled as a (yet to be pruned) Baseband Volterra Series consisting of odd order nonlinearities up to 9^{th} order. In terms of Question 2, we know from our *Statement of Research* that predistortion filter parameter estimation will be based on the *Spectral Power Feedback Learning (SPFL)* strategy. In this chapter, we present a formal mathematical framework for this learning strategy and go on to propose a new spectral power feedback objective more suited to current and future wideband applications.

9.1 Mathematical Framework of SPFL Strategy

The *Spectral Power Feedback Learning (SPFL)* strategy, as presented in Figure 9.1, models predistortion filter parameter estimation as a generic single objective mathematical optimization problem. In the applied mathematics community, such a problem is formally defined as follows [200]:

> Given a set of variables $\boldsymbol{h}$ plus a dependent function of those variables $B(\boldsymbol{h}) = b$, the goal of single objective mathematical optimization is to find the optimal variable values $\boldsymbol{h}_o$ which minimize the dependent function; $B(\boldsymbol{h}_o) = b_{min}$. The dependent function to be minimized is referred to as the objective function.

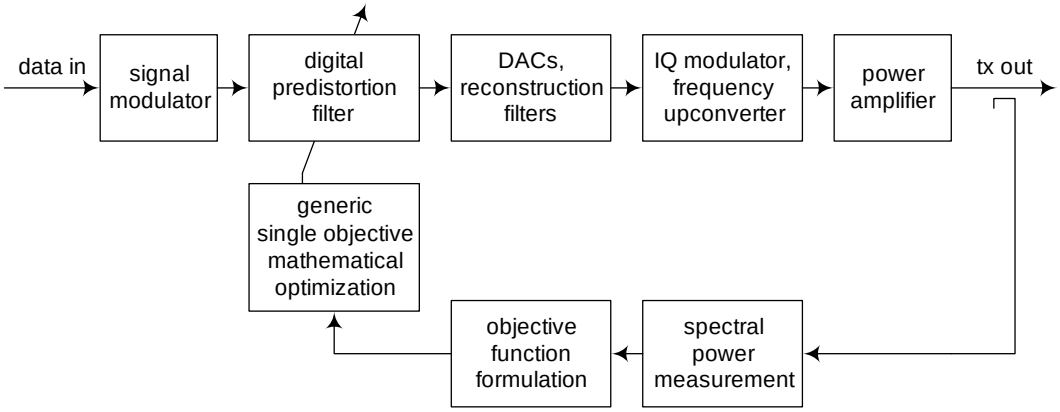


Figure 9.1: Predistortion filter parameter estimation based on the SPFL strategy

In accordance with this definition and in the context of the SPFL strategy:

- The predistortion filter's characterizing parameters are interpreted as the set of variables $\boldsymbol{h}$ to be optimized. Since the predistortion filter is modeled as the Baseband Volterra Series, these characterizing parameters are in fact a set of Volterra kernels. Specifically, with the predistortion filter containing only odd order nonlinearities up to 9^{th} order, the optimization vector space $\boldsymbol{h}$ can be represented as:

$$\boldsymbol{h} = \Big[\, \boldsymbol{p}_3 \;\big|\; \boldsymbol{p}_5 \;\big|\; \boldsymbol{p}_7 \;\big|\; \boldsymbol{p}_9 \,\Big] \tag{9.1}$$

  where for $m = 3, 5, 7, 9$:

$$\boldsymbol{p}_m = \Big\{\, p_m[i_1, \ldots, i_m] \;\; \forall \text{ causal } [i_1, \ldots, i_m] \,\Big\} \tag{9.2}$$

  Here, $p_m[i_1, \ldots, i_m]$ represents the m^{th} order predistortion filter kernel and causality implies nonnegative delay variables.[1]

- Since transmitter linearization is ultimately dependent on these predistortion filter kernels and hence $\boldsymbol{h}$, any single measure of output spectral distortion can be interpreted as the objective function $B(\boldsymbol{h}) = b$ to be minimized.

In summary, the optimal predistortion filter parameters $\boldsymbol{h}_o$ must be found which minimize the measure of spectral distortion, $B(\boldsymbol{h}_o) = b_{min}$, and ultimately linearize the transmitter. Here, the objective is assumed nonconvex with multiple local minima [78, 265]. Also, optimizations are performed numerically since the objective has no closed form; needing to be physically measured rather than formulated.

[1] (9.2) will be refined in Chapter 12 after the predistorter Baseband Volterra Series is pruned.

9.2 The New WACP Optimization Objective

In the literature to date, the SPFL strategy has only been reported with:

- linear signal modulation (QAM / QPSK)
- a very basic predistortion filter (up to 3 tunable parameters)
- the measure of spectral distortion being the power detected within a small bandpass region of the transmitter's adjacent channel [107, 262–265]

Since our application of the SPFL strategy is to wider band OFDM / CDMA systems exhibiting memory, our predistortion filter (specifically a pruned Baseband Volterra Series) will contain a greater number of tunable parameters than that reported above. With this increase in both modulation bandwidth and optimizer degrees of freedom comes the *regionalized* behavior of the adjacent channel distortion spectrum. By this it is meant that different regions of the adjacent channel distortion spectrum will generally behave differently during the predistortion process. For example, distortion reduction in one spectral region may be accompanied by no change, or even worse, distortion growth in another spectral region. This ultimately means that the *regionalized* measure of spectral distortion reported above is incapable of conveying complete adjacent channel behavior and in fact encourages the optimizer to tune the predistorter to solely reduce power in that region only; an unsatisfactory result. It follows that with this push to wider band applications comes the need for a more sophisticated multi-region distortion measure.

The multi-region distortion measure that we develop and propose in this research is called the *Weighted Adjacent Channel Power (WACP)* and is presented below:

$$\text{WACP} = \int_{\text{LAC}} W(f)P(f)\,df + \int_{\text{UAC}} W(f)P(f)\,df \tag{9.3}$$

Here, $W(f)$ represents a nonnegative, frequency dependent weighting function, specifically set to be a nonincreasing function of $|f - f_E|$ where f_E is the closest (upper or lower) transmission band Edge frequency. $P(f)$ represents the transmitter output Power Spectral Density as a function of frequency and LAC / UAC represent the integration domains of the Lower / Upper Adjacent Channel frequencies respectively. It can be seen that standard Adjacent Channel Power (ACP) is just the specific case of WACP for the constant weighting function $W(f) = 1$.

We propose the WACP objective over the more obvious standard ACP due to the inaccuracies of the predistortion filter model. That is, if one uses a primitive Volterra Series pruning strategy[2] or incorrectly estimates predistortion filter memory, then

[2]Pruning strategies will be discussed in Chapter 12

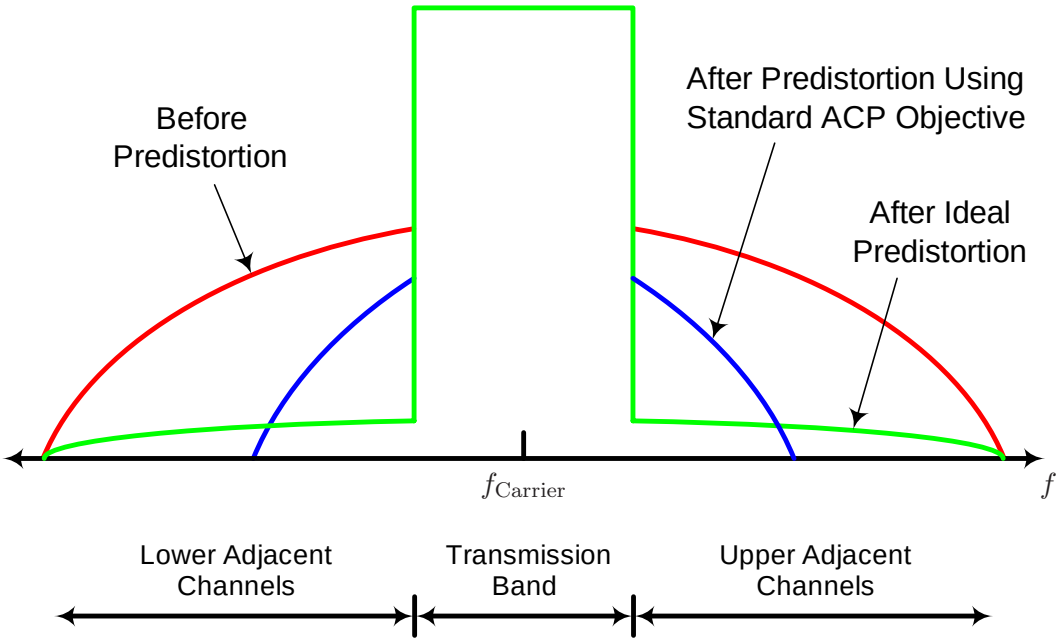


Figure 9.2: Power amplifier output spectra, before predistortion (red), after predistortion using a standard ACP objective (blue) and after ideal predistortion (green)

an optimizer using the standard ACP objective will tend to reduce outer adjacent channel distortion in favor of co-channel and inner adjacent channel distortion. As depicted in Figure 9.2, this is an unfavorable outcome since the final output mask filter cannot be relied upon to remove distortion close to band edges given its finite rolloff, plus co-channel distortion degrades BER performance. In either case, the optimizer simply cannot cope with the inaccuracies of the predistortion filter model. This behavior was experienced first hand in our very early experiments carried out on the laboratory transmitter testbed. Although predistortion filter model inaccuracies can be mitigated with appropriate design, they can never be totally eliminated and hence an ACP objective function is considered too unreliable.

In theory, adding the frequency dependent weighting function $W(f)$ to the standard ACP accumulation gives the objective function the added ability to discriminate between spectral distortion components, rather than treating them all equally in the accumulation. Specifically setting $W(f)$ to be a nonincreasing function of $|f - f_E|$ (spectral distance from the closest transmission band edge) then forces the optimizer to place greater emphasis on those previously neglected inner frequency components. In essence, the optimizer becomes more robust in the presence of predistortion filter model inaccuracies.

Many frequency dependent weighting functions $W(f)$ that fit the requirement of being nonincreasing functions of $|f - f_E|$ can be derived, the two simplest being the

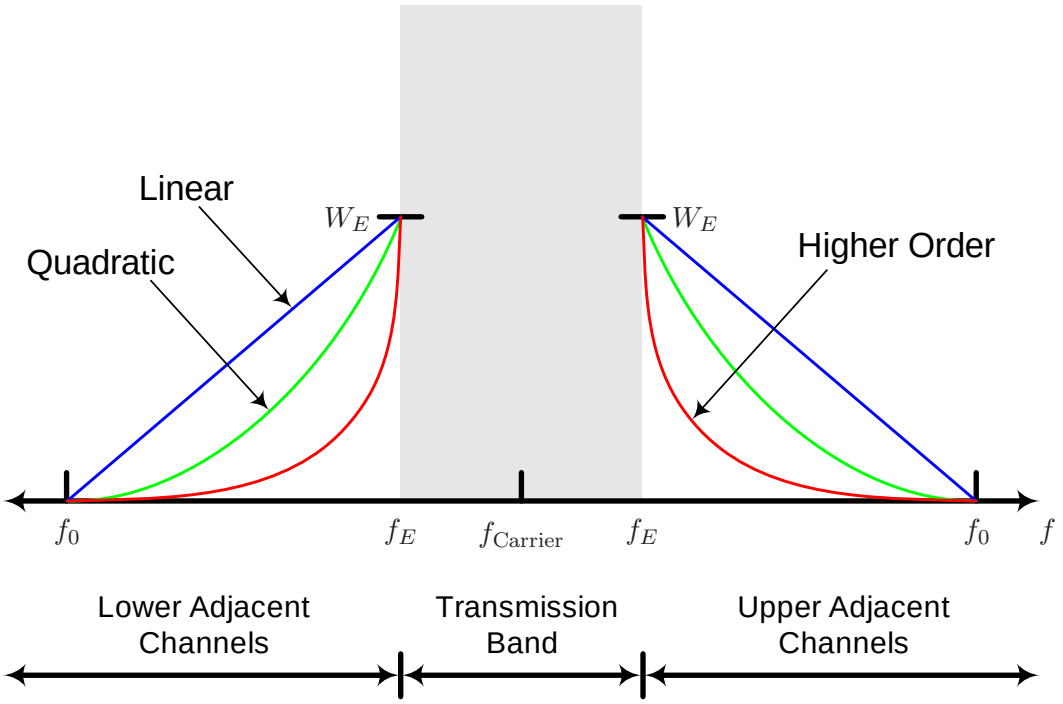


Figure 9.3: Linear, quadratic and higher order weighting functions plotted with respect to the allocated transmission band.

linear and quadratic functions which are formulated in (9.4) and (9.5) respectively:

$$W(f) = \begin{cases} 0 & \text{for } |f - f_E| > |f_0 - f_E| \\ \left(\dfrac{-W_E\,|f - f_E|}{|f_0 - f_E|}\right) + W_E & \text{for } |f - f_E| \leq |f_0 - f_E| \end{cases} \tag{9.4}$$

$$W(f) = \begin{cases} 0 & \text{for } |f - f_E| > |f_0 - f_E| \\ \left(\dfrac{W_E\,|f - f_E|^2}{|f_0 - f_E|^2}\right) - \left(\dfrac{2\,W_E\,|f - f_E|}{|f_0 - f_E|}\right) + W_E & \text{for } |f - f_E| \leq |f_0 - f_E| \end{cases} \tag{9.5}$$

For graphical reference, these functions are presented in Figure 9.3, along with an indicative higher order weighting function, demonstrating their relationship with respect to the allocated transmission band. As can be seen, W_E represents the weighting function amplitude at the allocated transmission band edge frequency f_E whilst f_0 represents the adjacent channel frequency at which the weighting falls to zero. In accordance with the integration domains of (9.3), $W(f)$ is not defined within the allocated transmission band as spectral distortion power $P(f)$ cannot be

measured here. In all situations, f_0 must be chosen to span all distortion components appearing above the indicated noise floor.

In practice, choosing a suitable weighting function amplitude W_E is a balance between the two extremes of *under-weighting* and *over-weighting*:

- *Under-weighting* is the scenario that arises when the weighting function amplitude W_E is insufficient and the optimizer behaves as if no weighting function exists at all. That is, identical to using a standard ACP objective, the optimizer tends to reduce outer adjacent channel distortion in favor of co-channel and inner adjacent channel distortion.

- *Over-weighting*, in direct contrast, is the scenario that arises when the weighting function amplitude W_E is excessive to the point where the optimizer tends to reduce inner adjacent channel distortion in favor of co-channel and outer adjacent channel distortion. This is undesirable with co-channel distortion degrading BER performance.

Both scenarios are presented in Figure 9.4. As one would expect, the initial stages of predistortion are particularly sensitive to *under-weighting* since inner adjacent channel distortion already dominates outer adjacent channel distortion. In direct contrast, the latter stages of optimization are particularly sensitive to *over-weighting* since inner and outer adjacent channel distortion is generally comparable at this

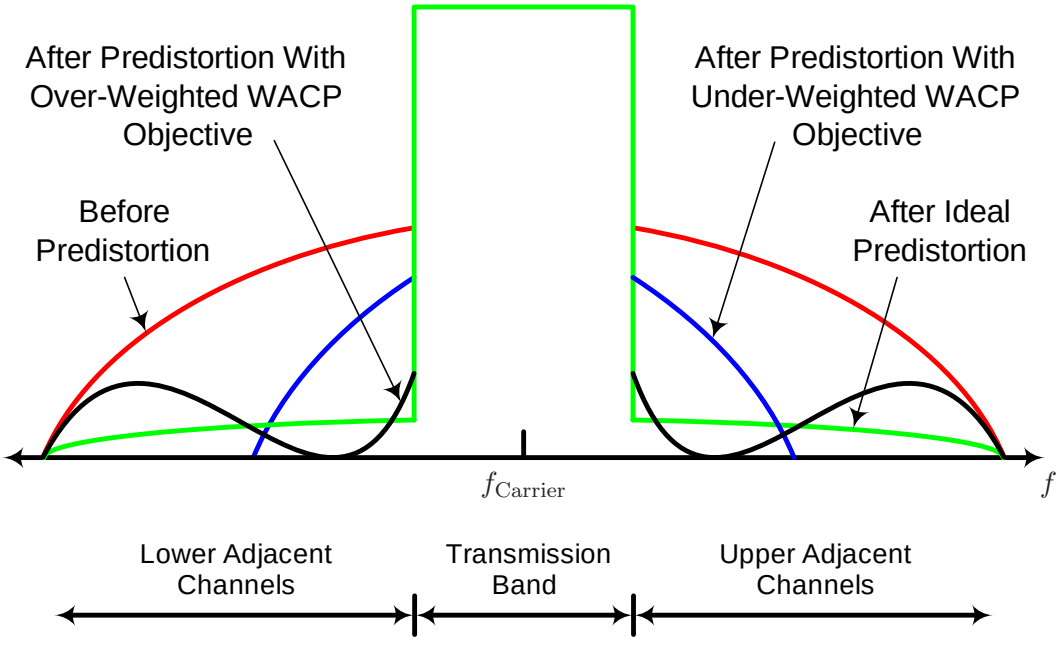


Figure 9.4: Power amplifier output spectra, before predistortion (red), after predistortion with over-weighted WACP objective (black), after predistortion with under-weighted WACP objective (blue) and after ideal predistortion (green).

stage. Taking this changing sensitivity into account, we propose the use of *weighting function taper*, the process of successively reducing the weighting function amplitude W_E throughout the optimization process. Compared to a fixed W_E, a tapered W_E prevents the occurrence of these *under* and *over-weighting* scenarios.

While many weighting functions and associated tapers can be defined, a quadratic weighting function (9.5) with initial $W_E = 100\,000$ and 10% linear taper for each optimization subphase thereafter, was found to give superior performance in our work[3]. This specific weighting function and taper is applicable to all DVB-T, WCDMA and DAB target modulations.

Several considerations must be taken into account when physically computing the WACP objective:

- Since a continuous frequency integration cannot be performed in practice, (9.3) must be discretized with respect to frequency:

$$\text{WACP} = \sum_{\text{LAC}\,|\,\Delta f} W(f)P(f) \;+\; \sum_{\text{UAC}\,|\,\Delta f} W(f)P(f) \tag{9.6}$$

  The discrete frequency step size Δf is chosen as a trade off between accumulation speed and WACP fidelity. $\Delta f = B/60$ was used with good effect in the experimental analysis of this research. Being a function of modulation bandwidth B, this step size applies to all of our target applications.

- (9.6) can be computed either via a DSP / FPGA implemented spectral estimation algorithm or a software controlled spectrum analyzer. Based purely on hardware availability, we choose the latter. With appropriate measurement settings [73, 286, 287], the analyzer is instructed to sequentially sweep its marker to each discrete frequency and report the $P(f)$ measurement. A subroutine then performs the final weighted accumulation.

  On the laboratory transmitter testbed, this software procedure is implemented by the function *MeasureObjectiveFunction()*. To be precise, this is a member function of the object-oriented class *ObjectiveFunction ACS*. Corresponding function-definition and class-declaration source code resides in project files *ObjectiveFunction ACS Templates.cpp* and *ObjectiveFunction ACS.h* respectively. Both files are located within folder *Software\Cpp* on the accompanying DVD. Spectrum analyzer measurement settings were discussed in Section 4.5.

[3]Optimization schedules and subphases will be discussed further in Chapter 10

- Technically speaking, our target modulating signals (DAB, WCDMA, DVB-T) are random processes given the inherent randomness of the input data stream. It follows that the transmitter output signal is also a random process and the WACP objective (9.6) is theoretically a random variable with mean and spread.

  In this respect, WACP averaging will logically provide a more reliable objective estimate and hence aid optimizer convergence. However, considering the spectrum analyzer's multi-second sweep time[4], this averaging also leads to slow optimizer tracking. Fortunately, by choosing robust optimization algorithms (to be discussed in Chapter 11), the detrimental convergence effects of WACP randomness can be mitigated and the amount of averaging reduced to zero.

Apart from a Volterra Series pruning refinement of (9.2), we have now formally defined the optimization framework of the SPFL strategy in terms of the new WACP objective. This framework sets the foundation for all future optimization activities.

To summarize, this chapter has discussed the fundamental aspects of predistortion filter parameter estimation. Specifically, a new objective function called the Weighted Adjacent Channel Power (WACP) has been proposed for the SPFL strategy. Compared with traditional spot power objectives, WACP is more suited to current and future wideband applications based on its ability to convey complete adjacent channel behavior and furthermore discriminate between spectral distortion components.

[4]A function of resolution and video bandwidth

Chapter 10

Mathematical Optimization Process

With the optimization framework now formally defined, attention is turned to the optimization process. Despite the mathematical elegance of the optimization framework, computing the optimal $\boldsymbol{h}_o$ is not as simple as performing a single, one-off optimization over the entire vector space $\boldsymbol{h} = \left[\boldsymbol{p}_3 \mid \boldsymbol{p}_5 \mid \boldsymbol{p}_7 \mid \boldsymbol{p}_9 \right]$. In this chapter we show that practical factors, specifically *transmitter nonlinear drift* and *optimization convergence reliability*, demand a more complex optimization process defined in terms of *phases* and *schedules* respectively.

10.1 Two Distinct Phases of Optimization

From an operational perspective, two distinct phases of optimization must be defined, these being the *Initial Setting* and *On-Air Adaption* phases.

- The *Initial Setting* phase estimates the optimal predistortion filter parameters $\boldsymbol{h}_o$ at the very start of the transmitter's operational life. This phase occurs with the transmitter output switched to a dummy load as the regulatory spectral mask [41] will not be met until the predistortion filter parameters approach optimality. Once the *Initial Setting* phase is complete and hence the regulatory mask is met, the transmitter is ready for broadcast and its output can be switched to the antenna.

- In practice, a transmitter's nonlinear transfer characteristic will drift slowly during its operational life. This is a result of component aging (transistors and capacitors), temperature fluctuations and power supply voltage variations [50, 107, 288]. This ultimately means that the predistortion filter parameters estimated during the *Initial Setting* phase do not remain optimal over

the entire lifetime of the transmitter, hence the need for an *On-Air Adaption* phase. The *On-Air Adaption* phase adapts the predistortion filter parameters in order to maintain optimality whilst the transmitter's nonlinear transfer characteristic is changing. This phase must occur whilst the transmitter is on-air, performing its intended communication function, since taking the transmitter off-air is both undesirable and costly for the transmitter owner.

So we see that the optimization process is not a one-off event, as one might initially assume, but rather an on-going activity performed throughout the transmitter's operational life.

10.2 Maximal Convergence Reliability

How we design the optimization processes for each of these *Initial Setting* and *On-Air Adaption* phases is driven by the requirement for maximal convergence reliability. Generally speaking, common reasons for poor convergence reliability are as follows [54, 74, 200]:

1. An excessively large variable vector $\boldsymbol{h}$ and therefore impractically vast search space.
2. Large differences in the magnitude of variables and therefore adverse variable sensitivity. This is commonly referred to as *poor scaling*.
3. A complex objective function surface exhibiting multiple inferior local minima.
4. Inability of an optimizer to hone in on a minimum once it has been located.
5. Premature exiting of an optimizer.
6. Poor estimation of Gradient vector or Hessian matrix parameters.

Armed with this knowledge, maximal convergence reliability can be achieved in our optimization processes by employing the following respective Counter-Measures whenever and wherever possible:

CM 1. Minimize the number of variables per optimization:

- **A.** Only include influential variables in the optimization process at all times.
- **B.** Optimize real and imaginary parts of complex variables separately.
- **C.** Prune the predistortion filter Baseband Volterra kernel.

CM 2. Pre-scale variables with respect to nonlinear order. Experimental analysis indicates poor scaling, with an average 40 dB reduction in variable magnitude per odd nonlinear order. Also avoid *stepping* optimizers since step size estimation is typically unreliable when variables are poorly scaled.

CM 3. Use global optimization, at least to regionalize the true global minimum and screen out inferior local minima.

CM 4. Perform precision optimization:

A. Regularly reset optimizers with fresh initialization parameters to avoid *stagnation*.

B. Use a local optimizer to hone in on the minimum identified by a global optimizer; the reason being, local optimizers tend to be more nimble than global optimizers.

CM 5. Develop a reliable means of detecting when an optimizer has achieved its goal; whether it be to pinpoint or just regionalize a minimum.

CM 6. Consider simpler but more robust gradient-free optimization algorithms as an alternative to popular gradient based algorithms.

In essence, the requirement for maximal convergence reliability eliminates any perceived weaknesses in the SPFL strategy's parameter estimation process.

10.3 Optimization Scheduling

While running a single optimization over the entire vector space is considered the simplest optimization process for the *Initial Setting* and *On-Air Adaption* phases, it is also the optimization process which exhibits the minimal convergence reliability for two reasons. Firstly, it employs the largest possible variable vector which, according to Dot-Point 1 of the previous section, places it at higher risk of poor convergence. Secondly, the variable vector $\boldsymbol{h} = \left[\boldsymbol{p}_3 \mid \boldsymbol{p}_5 \mid \boldsymbol{p}_7 \mid \boldsymbol{p}_9 \right]$ exhibits maximally poor scaling which, according to Dot-Point 2 of the previous section, also places it at higher risk of poor convergence.

It now becomes obvious that the optimization process which achieves maximal convergence reliability is one that is based on a set of sequential optimizations over subsets $\boldsymbol{h}_s$ of the entire vector space $\boldsymbol{h}$. This set of optimizations will from here on be referred to as an *optimization schedule*.

When developing these optimization schedules, one must adhere to the *ascending nonlinear order* principle first reported in Section 6.3 and alluded to thereafter. This concept relates to the order in which predistortion filter kernels are to be computed.

Optimization Number	**Vector Subspace $\boldsymbol{h}_s$**	**Objective Function**
1^{st} Optimization	$\boldsymbol{p}_3$	WACP
2^{nd} Optimization	$\boldsymbol{p}_5$	WACP
3^{rd} Optimization	$\boldsymbol{p}_7$	WACP
4^{th} Optimization	$\boldsymbol{p}_9$	WACP

Table 10.1: Simplest schedule adhering to the *ascending nonlinear order* principle

Specifically, if parasitic growth is to be fully accounted for during the linearization process, predistortion must be performed in ascending nonlinear order. That is, for the specific case of optimization, kernel coefficients $\boldsymbol{p}_m$ must be optimized prior to, or in conjunction with $\boldsymbol{p}_{m+2}$ for $m = 3, 5, 7, 9$.

The simplest schedule that adheres to this principle is presented in Table 10.1. Here, a separate optimization is allocated to each predistortion filter kernel order. In addition to being simple, this schedule benefits from minimally sized optimization vectors and good scaling. Based on these properties, one would consider this to be the schedule with maximal convergence reliability. However this is not the case for the following reason.

As discussed in Section 6.3, $\boldsymbol{p}_m$ is tuned specifically to remove m^{th} order distortion. In this sense, m^{th} order distortion must be observable at all times if the tuning is to be successfully controlled. This observability cannot be guaranteed however since all orders of distortion are collocated in frequency, specifically around the carrier, and resultant distortion is what is actually observed. Once the level of m^{th} order distortion falls below that of all other orders of distortion, observability is lost and tuning of $\boldsymbol{p}_m$ fails. Based on this *observability* issue, we conclude that optimization schedules cannot be based solely on the individual nonlinear order level.

The optimization schedule that we propose is based on counter-measure CM 1-A and the idea of setting $\boldsymbol{h}_s$ to be the *influential* subset of $\boldsymbol{h}$ at all times. Here, we define the *influential* subset as that which has an immediate influence on reducing the WACP objective. That is, if $m_1 \dots m_x$ are the currently dominant nonlinear orders of distortion forming the WACP objective, then the current *influential* subset is defined as $\left[\, \boldsymbol{p}_{m_1} \,\middle|\, \dots \,\middle|\, \boldsymbol{p}_{m_x} \,\right]$ since $\boldsymbol{p}_m$ is primarily responsible for reducing m^{th} order distortion. This idea is summarized in Table 10.2.

Step 1	Identify current *influential* subset of $\boldsymbol{h}$ based on the WACP's dominant distortion components
Step 2	Set $\boldsymbol{h}_s$ to *influential* subset
Step 3	Optimize over $\boldsymbol{h}_s$

Repeat above steps until WACP objective is minimized

Table 10.2: Summarizing steps of proposed optimization schedule

By adhering to this strategy, $\boldsymbol{h}_s$ will always be the minimally sized variable vector which guarantees complete observability and for this reason, the resulting schedule will exhibit maximal convergence reliability.

In practice, the *influential* subset does not need to be identified in real-time as suggested in Table 10.2. It can in fact be predicted based on our previous Chapter 6 & 7 analysis. In the following sections, we perform these predictions and hence develop precise, working optimization schedules for the *Initial Setting* and *On-Air Adaption* phases. As will be seen, the proposed schedules do indeed satisfy the *ascending nonlinear order* principle due to the inverse relationship between nonlinear order and distortion power.

10.4 *Initial Setting* Optimization Schedule

Consider the scenario at the commencement of the *Initial Setting* phase. Predistortion is yet to be applied and therefore output distortion is purely amplifier generated. In general, a power amplifier's 3^{rd} order nonlinearity will be significantly larger than its higher order nonlinearities, as previously demonstrated in Figure 7.2 (page 64). If follows that the WACP objective will be dominated by 3^{rd} order distortion and hence the *influential* subset can be defined as $\boldsymbol{p}_3$. According to Table 10.2, the first scheduled optimization must therefore be performed over $\boldsymbol{h}_s = \boldsymbol{p}_3$. To the point, including higher order kernels in this first optimization would have no effect other than to jeopardize convergence.

Whilst 3^{rd} order distortion will reduce during this initial optimization, parasitic nonlinearities generated by the dominant 3^{rd} order amplifier nonlinearity will cause 5^{th}, 7^{th} and 9^{th} order distortion to grow significantly, as outlined in Chapter 8. It follows that with 3^{rd} order distortion reducing and 5^{th}, 7^{th} and 9^{th} order distortion growing, a converging point will be reached whereby these orders of distortion become comparable. With a further $3-6\,\text{dB}$ reduction in 3^{rd} order distortion, observability completely diminishes and the optimizer will stall, signifying the end of the initial optimization.

The WACP objective will then be dominated by 5^{th}, 7^{th} and 9^{th} order distortion and hence the *influential* subset can be defined as $\left[\, \boldsymbol{p}_5 \mid \boldsymbol{p}_7 \mid \boldsymbol{p}_9 \,\right]$. According to Table 10.2, the second scheduled optimization must therefore be performed over $\boldsymbol{h}_s = \left[\, \boldsymbol{p}_5 \mid \boldsymbol{p}_7 \mid \boldsymbol{p}_9 \,\right]$. During this optimization, 5^{th}, 7^{th} and 9^{th} order distortion will reduce while 3^{rd} order distortion will remain fixed, as outlined in Chapter 6. It follows that a point will again be reached whereby these orders of distortion become comparable. With a further 3 – 6 dB combined reduction in 5^{th}, 7^{th} and 9^{th} order distortion, observability completely diminishes and the optimizer will stall, signifying the end of the second optimization.

At this point, the WACP objective is once again dominated by 3^{rd} order distortion and therefore the scenario is similar to the beginning of the first scheduled optimization; just not as pronounced. This observation logically suggests that the overall optimization schedule must be a repeating cycle of these first two scheduled optimizations. With each cycle comes further fine tuning of the vector space estimate. Whilst multiple repetitions can be performed, testing indicates that only one repetition is satisfactory for rounding out the optimization schedule. As a result, the third and fourth scheduled optimizations can be defined over $\boldsymbol{h}_s = \boldsymbol{p}_3$ and $\boldsymbol{h}_s = \left[\, \boldsymbol{p}_5 \mid \boldsymbol{p}_7 \mid \boldsymbol{p}_9 \,\right]$ respectively.

In terms of the WACP objective, we propose using the *quadratic* weighting function (9.5) with initial $W_E = 100\,000$ and 10% linear taper for each scheduled optimization thereafter. Of all the different functions and tapers tested with respect to the *under-* and *over-weighting* phenomena discussed in Section 9.2, this combination stands out as being the simplest and most reliable across all of our target signal modulations.

In accordance with counter-measure CM 3 of Section 10.2, a global optimization algorithm must be employed for the first and second scheduled optimizations in order to locate the true global minimum of the assumed nonconvex objective. However, in accordance with counter-measure CM 4-B, a more nimble local optimization algorithm may be used for the third and fourth scheduled optimizations since the search space will have already been narrowed to the vicinity of the global minimum and a local refinement is all that is required. Precise algorithms to implement these global and local optimizations will be discussed in the next chapter.

Finally, in accordance with counter-measure CM 1-B, a simple but effective measure to further reduce the number of variables per optimization, hence improve optimization convergence reliability, is to optimize each above defined subset $\boldsymbol{h}_s$ over its real and imaginary components separately. In this context, real components are optimized first followed by imaginary components. This is because imaginary components are generally an order of magnitude smaller than real components; as determined during experimental analysis.

Optimization	$\boldsymbol{h}_s$	Optimizer	WACP Weighting & Taper
1a.	$\boldsymbol{p}_{3R}$	Global	Quadratic, $W_E = 10^5$
1b.	$\boldsymbol{p}_{3I}$		
2a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		Quadratic, $W_E = 10^4$
2b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		
3a.	$\boldsymbol{p}_{3R}$	Local	Quadratic, $W_E = 10^3$
3b.	$\boldsymbol{p}_{3I}$		
4a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		Quadratic, $W_E = 10^2$
4b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		

Table 10.3: Proposed *Initial Setting* optimization schedule

Based on the above analysis, the proposed optimization schedule for the *Initial Setting* phase is presented in Table 10.3. Here, vector subscripts R and I signify real and imaginary components respectively. Figure 10.1 goes on to demonstrate the superior performance of this schedule when compared to a single global optimization over the entire vector space $\boldsymbol{h}$.

10.5 *On-Air Adaption* Optimization Schedule

As outlined in Section 10.1, the *On-Air Adaption* phase must adapt predistortion filter parameters in order to maintain optimality whilst the transmitter's nonlinear transfer characteristic is drifting. While nonlinear drift will cause all orders of distortion to grow, individual growth rates will not be the same. To be precise, growth in 3^{rd} order distortion will be most severe due to the power amplifier's 3^{rd} order dominance. This is demonstrated in Figure 10.2 for progressively larger forced drifts[1] following the *Initial Setting* phase. In all cases, distortion growth is concentrated close to transmission band edges; a clear indication of dominant 3^{rd} order presence.

It follows that after any drift activity, the level of 3^{rd} order distortion will rise above that of comparable 5^{th}, 7^{th} and 9^{th} order distortion and hence the optimization scenario is similar to the beginning of the third scheduled *Initial Setting* optimization discussed in the previous section. This observation logically suggests that, in terms of vector subsets $\boldsymbol{h}_s$, the associated *On-Air Adaption* schedule must be a replica of the second half of the *Initial Setting* phase.

[1] *forced drift* is caused by deliberate changes in amplifier supply voltage & ambient temperature

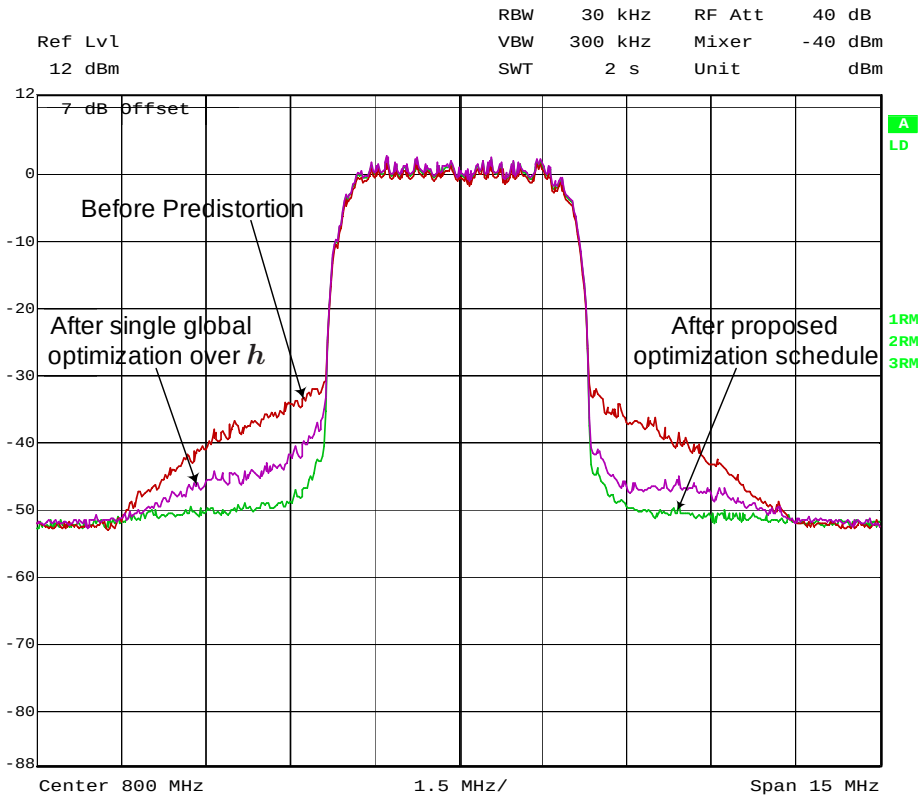


Figure 10.1: Power spectra before predistortion (red), after *Initial Setting* via a single global optimization over $\boldsymbol{h}$ (magenta) and after *Initial Setting* via the proposed optimization schedule (green) for a real 25 Watt Class-AB push-pull power amplifier with WCDMA signal modulation.

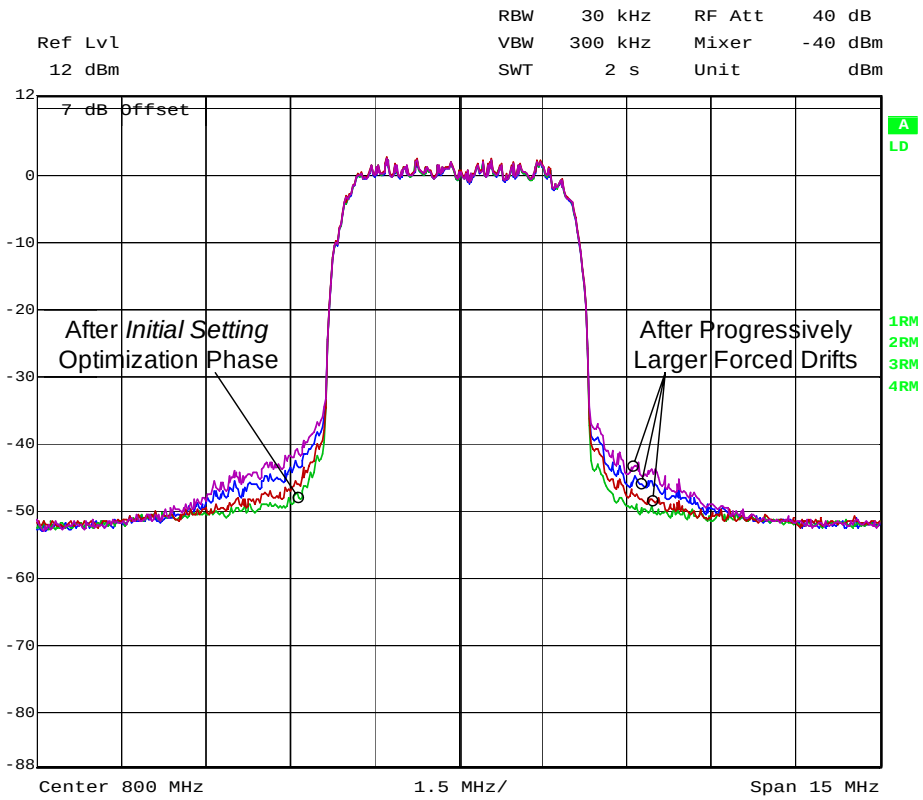


Figure 10.2: Power spectra for progressively larger forced drifts following the *Initial Setting* phase. Forced drifts caused by deliberate changes in power amplifier supply voltage and ambient temperature. Real 25 Watt Class-AB push-pull power amplifier with WCDMA signal modulation.

The *On-Air Adaption* phase should commence immediately following the *Initial Setting* phase and repeat indefinitely for the lifetime of the transmitter. In doing so, the drifting objective function minimum will be under constant track and re-convergence reliability is maximal.

As outlined in Section 10.1, nonlinear drift is attributed to component aging, temperature fluctuations and power supply voltage variations.

- Since component aging is a continuous process, it can be assumed to cause the objective function minimum to translate in the vector space rather than abruptly disappear / reappear elsewhere. This translation suggests the use of local rather than global optimization for tracking the minimum.

- Since temperature and power supply variations are centered about a mean, they can be assumed to cause the objective function minimum to move back and forth about the point corresponding to the mean state. This oscillation also suggests the use of local rather than global optimization.

Precise algorithms to implement these proposed local optimizations will be discussed in the next chapter. Irrespective of the chosen algorithms however, fine search movements must be utilized to avoid tracking jitter during periods of little drift.

In terms of the WACP objective, we propose a *quadratic* weighting function with fixed $W_E = 100$. This is a continuation of the weighting utilized in the last scheduled optimization of the *Initial Setting* phase. In practice, the lower levels of distortion growth caused by natural nonlinear drift do not warrant weighting function taper.

Finally, as was the case for the *Initial Setting* phase, each defined subset $\boldsymbol{h}_s$ is optimized over its real and imaginary components separately in order to reduce the number of variables per optimization and hence improve re-convergence reliability.

Based on the above analysis, the proposed optimization schedule for the *On-Air Adaption* phase is presented in Table 10.4.

Optimization	$\boldsymbol{h}_s$	**Optimizer**	**WACP Weighting**
1a.	$\boldsymbol{p}_{3R}$		
1b.	$\boldsymbol{p}_{3I}$	Local	Quadratic, $W_E = 10^2$
2a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		
2b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		
	Repeat indefinitely for constant tracking		

Table 10.4: Proposed *On-Air Adaption* optimization schedule

By designing the Initial Setting and On-Air Adaption schedules according to the influential subset strategy, maximal convergence reliability is guaranteed. This is because optimization is always restricted to the minimally sized variable vector for which observability holds. From a high level perspective, the resulting schedules are simply a reflection of the optimizer needing to switch between the two tasks of removing dominant distortion (3^{rd} order) and cleaning up those generated parasitics (5^{th}, 7^{th} and 9^{th} order).

Chapter 11

Mathematical Optimization Algorithms

In the previous chapter, individual optimizations of the *Initial Setting* and *On-Air Adaption* schedules were assigned local / global classifications. In this chapter, we take the next step and assign specific optimization algorithms. With mathematical optimization being a mature field of applied mathematics, this assignment task involves assessing the practical suitability of the many existing algorithms, in accordance with the properties of our predistortion application.

11.1 Algorithm Classifications

Within the applied mathematics literature, the following algorithm classifications apply:

- Single vs Multi-Objective
- Constrained vs Unconstrained
- Discrete vs Continuous
- Gradient vs Gradient-Free
- Stochastic vs Deterministic

These classifications are in addition to the already familiar *local vs global* classification. In this section, we review these classifications in the context of our optimization framework, with the goal of understanding what type of algorithm we are actually seeking.

Single vs Multi-Objective

If an optimization problem is defined with a single objective function $B(\boldsymbol{h})$ then the problem is called a *single objective* optimization problem. The goal is to find the optimal vector $\boldsymbol{h}_o$ which minimizes the objective.

If on the other hand an optimization problem is defined with multiple, possibly conflicting objective functions $[\, B_1(\boldsymbol{h})\ B_2(\boldsymbol{h})\ \ldots\ B_x(\boldsymbol{h})\,]$, then the problem is called a *multi-objective* optimization problem. Given two or more of the objectives are in general conflicting, there won't exist a single optimal vector which simultaneously minimizes each objective. The goal in this case is to find the optimization vector which optimally trades off each objective according to some predefined criteria.

It is also possible to convert a *multi-objective* problem into a *single objective* problem by weighting each objective and accumulating as follows:

$$B(\boldsymbol{h}) = W_1\, B_1(\boldsymbol{h}) + W_2\, B_2(\boldsymbol{h}) + \ldots + W_x\, B_x(\boldsymbol{h}) \tag{11.1}$$

In such cases, $B(\boldsymbol{h})$ is specifically known as an *aggregated* single objective function and the optimal solution $\boldsymbol{h}_o$ depends on the relative values of the specified weights [246, 266].

By definition, WACP is considered an *aggregated* single objective function since it is the weighted accumulation of multiple objectives; these objectives being the spectral powers of each adjacent channel frequency:

$$\text{WACP} = \int_{\text{LAC}} W(f)P(f)\,df + \int_{\text{UAC}} W(f)P(f)\,df \tag{11.2}$$

Irrespective of whether WACP is *pure* or *aggregated*, our optimization problem is by definition *single objective* and must be solved using single objective optimization algorithms.

Constrained vs Unconstrained

Constrained optimization problems are those which have constraints placed on their optimization vector space elements. These constraints may be simple upper / lower bounds on individual elements or more complex inter-element linear / nonlinear, equalities / inequalities representing physical relationships among the elements [200]. *Unconstrained* optimization problems on the other hand are those which don't possess vector space element constraints. Different optimization algorithms must be used depending on whether the problem is *constrained* or *unconstrained*.

In terms of our optimization framework, the vector space $\boldsymbol{h}$ is defined as the entire set of predistortion filter kernel coefficients. These kernel coefficients are unbounded and can be considered unrelated to each other in the context of optimization. In

this sense, no constraints exist on the vector space elements. It follows that our optimization problem is by definition *unconstrained* and must be solved using an unconstrained optimization algorithm.

Discrete vs Continuous

In a *discrete* optimization problem, elements of the optimization vector space can only take on discrete values and therefore the vector space is considered a finite set. This is in direct contrast to a *continuous* optimization problem whereby elements of the optimization vector space can take on continuous values and therefore the optimization vector space is considered an infinite set. Different optimization algorithms are used based on whether the problem is *discrete* or *continuous*.

In terms of our optimization framework, the vector space $\boldsymbol{h}$ is defined as the entire set of predistortion filter kernel coefficients. These kernel coefficients are continuous by nature taking on values from the complex number plane $\boldsymbol{\mathcal{C}}$. In this sense, the optimization vector space is the infinite complex set $\boldsymbol{\mathcal{C}}^S$ where S is the size of $\boldsymbol{h}$. It follows that our optimization problem is by definition *continuous* and must be solved using a continuous optimization algorithm.

Gradient vs Gradient-Free

Gradient based optimization algorithms utilize the objective function's first and second order derivative characteristics (Gradient vector and Hessian matrix respectively) to navigate to the local / global solution. In practice, these characteristics are approximated via *Finite Differences* and *Symmetric Rank 1* updating [200]. Whilst gradient based optimization algorithms are technically superior to other forms of optimization, they are known to be computationally intensive and susceptible to measurement noise and vector space scaling issues.

In direct contrast, *gradient-free* optimization algorithms rely solely on objective function measurements to navigate to the local / global solution. They do not require knowledge of the objective function's derivative characteristics. Whilst gradient-free optimization algorithms are not as technically apt as gradient based algorithms, they are known to be more robust and computationally efficient.

Given both *gradient* and *gradient-free* algorithms have their unique advantages, it would be wise at this stage to consider both types as candidate algorithms.

Stochastic vs Deterministic

Stochastic optimization algorithms are those which utilize probabilistic or random decision making. At each execution step of a stochastic algorithm, there potentially exists more than one way to proceed. This is in direct contrast to *deterministic*

algorithms which possess a clear decision making process. At each execution step of a deterministic algorithm, there exists only one way to proceed.

Deterministic algorithms are known to be thorough in their search for local and global minima. For low and medium dimensionality problems, this is an advantage. However for high dimensionality problems, this thoroughness leads to long run times. In such cases, *stochastic* algorithms are favored as they trade in guaranteed correctness of the solution for shorter run times [283].

Given both *deterministic* and *stochastic* algorithms have their unique advantages, it would be wise at this stage to consider both types as candidate algorithms.

Based on the above review of algorithm classifications, we conclude that *single objective*, *unconstrained*, *continuous*, *local* and *global* optimizers are required to solve our optimization problem. These optimizers can be either *gradient* or *gradient-free* as well as either *stochastic* or *deterministic*.

11.2 Theoretical Shortlist of Optimization Algorithms

A search of the applied mathematics literature for algorithms fitting this criteria identifies the following theoretical shortlist of optimization algorithms:

Optimization Algorithm	**Local or Global**	**Gradient or Gradient-Free**	**Stochastic or Deterministic**
Gradient Descent	L	G	D
Trust Region Newton	L	G	D
Alpha Branch & Bound	G	G	D
Nelder-Mead Simplex	L	GF	D
Genetic	G	GF	S

Table 11.1: Theoretical shortlist of optimization algorithms

It is noted that the *Simulated Annealing* algorithm (local, gradient-free, stochastic) also fits this criteria but fails to make the shortlist. This is because the algorithm's central control function, the *annealing schedule probability function*, is unknown for our predistortion application [141, 213]. If the *annealing schedule function* is set too conservatively, the algorithm will randomly bounce around the search space without convergence. If the *annealing schedule function* is set too aggressively, the

algorithm will revert to random steps in the direction of objective function reduction but with unknown convergence rate. Estimating the *annealing schedule probability function* requires experimental trial and error and hence the algorithm is considered too unreliable in the context of achieving maximal convergence reliability.

A complete treatise of Table 11.1's shortlisted algorithms is presented in Appendix A. Readers are strongly encouraged to consult this Appendix in order to familiarize themselves with the inner workings of each algorithm.

11.3 Practical Suitability of Shortlisted Algorithms

Leveraging on the theory and hardware implementation of Appendix A, we now proceed to examine the practical suitability of each shortlisted algorithm in accordance with the properties of our predistortion application. We start with local algorithms then proceed globally.

Gradient Descent (Local)

✓ The *deterministic* nature and hence clear decision making process of this algorithm is favorable.

✗ For even-numbered local optimizations of the *Initial Setting* and *On-Air Adaption* schedules, where $\boldsymbol{h}_s$ spans multiple orders ($5^{\text{th}} \,|\, 7^{\text{th}} \,|\, 9^{\text{th}}$), poor scaling will lead to step size sensitivity. As discussed in Section 10.2, experimental analysis indicates an average 40 dB reduction in variable magnitude per odd nonlinear order. In this sense, if step size is chosen to match the larger magnitude of lower order variables, higher order variables will become extremely sensitive causing dramatic changes in the objective function. If step size is chosen to match the smaller magnitude of higher order variables, little progress is made in optimizing the lower order variables. In practice it is very difficult to find an adequate tradeoff, even when employing variable pre-scaling.

✗ For odd-numbered local optimizations of the *Initial Setting* and *On-Air Adaption* schedules, where $\boldsymbol{h}_s$ is restricted to 3^{rd} order variables and scaling is good, estimation of the Gradient vector and hence step direction will be sensitive to WACP randomness. As discussed in Section 9.2, the WACP objective is theoretically a random variable with mean and spread. In theory, such a sensitivity problem could be overcome by averaging WACP measurements during the Gradient estimation process. However with:

- at least two seconds per raw WACP measurement
  ↪ determined by spectrum analyzer sweep, resolution/video BW

- at least 5 raw WACP measurements per averaged WACP measurement
  ↪ to adequately reduce variance
- at least $3n$ averaged WACP measurements per Gradient vector estimate[1]
  ↪ determined by the robust *least squares* technique of Appendix B.1

this averaging would lead to a very slow stepping / convergence rate and hence only be viable for small variable vectors.

Since we generally can't rely on the prospect of small variable vectors, especially for future wider band modulation strategies, this algorithm cannot be used for either even- or odd-numbered scheduled optimizations and hence is considered unsuitable for our predistortion application.

Trust Region Newton (Local)

✓ The *deterministic* nature and hence clear decision making process of this algorithm is favorable.

✓ This algorithm utilizes both first order (Gradient vector) and second order (Hessian matrix) derivative characteristics to model the objective function locally at each step. The domain of this model is further verified by an appropriate trust region. This high level of objective conformity, coupled with analytic computation of the model minimum, ensures poor scaling is not a problem.

✗ At each step, estimation of the Gradient vector / Hessian matrix and hence quadratic objective model will be sensitive to WACP randomness. As discussed in Section 9.2, the WACP objective is theoretically a random variable with mean and spread. This sensitivity will subsequently diminish the associated trust region and hence modeling domain, leading to reduced convergence rate and reliability. In theory, such a problem could be overcome by averaging WACP measurements during the Gradient / Hessian estimation processes. However, with:

- at least two seconds per raw WACP measurement
  ↪ determined by spectrum analyzer sweep, resolution/video BW
- at least 5 raw WACP measurements per averaged WACP measurement
  ↪ to adequately reduce variance

[1] n being the dimension of the $\boldsymbol{h}_s$ vector

- at least $3n$ averaged WACP measurements per Gradient vector estimate
  ↪ determined by the robust *least squares* technique of Appendix B.1
- at least $3n^2+3n$ averaged WACP measurements per Hessian matrix estimate
  ↪ determined by the robust *hybridized* technique of Appendix B.2

this averaging would lead to an excruciatingly slow stepping / convergence rate, worse than Gradient Descent due to additional Hessian and trust region estimation, and hence be practically unviable even for small variable vectors. This same conclusion is reached for *Symmetric-Rank-1 Updating* of the Hessian estimate.

Despite being resilient to poor scaling, this last ✗ renders the algorithm unsuitable for our predistortion application.

Nelder-Mead Simplex (Local)

✓ The *deterministic* nature and hence clear decision making process of this algorithm is favorable.

✓ Since all simplex geometric manoeuvres (reflection, contraction, expansion, shrinkage) are dimensionally orthogonal, pre-scaling variables with respect to nonlinear order during simplex initialization allows the algorithm to accommodate for poor scaling. This is in accordance with counter-measure CM 2 of Section 10.2. In practice, pre-scaling magnitude for each nonlinear order is set to be the variable increment size which induces a small but observable change in the output distortion spectrum. In accordance with CM 4, routine resetting of the simplex also ensures scaling induced *stagnation* is avoided.

✓ WACP randomness will only interfere with this *objective measurement and comparison* strategy when simplex vertices are all of the same or similar objective value. Generally speaking, this only occurs once the simplex has geometrically contracted within the desired local minimum and hence achieved its goal. It follows that this gradient-free algorithm can cope with WACP randomness without having to rely on WACP measurement averaging and is thus capable of large-scale optimization.

With its resilience to poor scaling and WACP randomness, this algorithm is considered highly robust and therefore well suited to our predistortion application.

Alpha Branch & Bound (Global)

✓ The *deterministic* nature and hence clear decision making process of this algorithm is favorable.

✓ For each boxed region, poor variable scaling will be transfered from the objective function to the convex under-estimating function $\mathcal{L}(\boldsymbol{h})$. This algorithm's ability to cope with poor variable scaling is thus determined by the local optimizer chosen to estimate each region's single $\mathcal{L}(\boldsymbol{h})$ minimum. In addition to poor scaling, this bounding local optimizer will experience inherited WACP randomness. As discussed above, the Nelder-Mead Simplex algorithm is resilient to this poor scaling and objective randomness, and is thus the perfect choice of local optimizer to ensure the ABB algorithm performs well in the presence of poor variable scaling.

✗ For each boxed region, spot Hessian matrix estimates will be sensitive to WACP randomness. This sensitivity will subsequently diminish the Hessian Interval matrix estimate and therefore convex under-estimating function $\mathcal{L}(\boldsymbol{h})$, leading to either over-bounding (slow convergence) or under-bounding (no convergence). In theory, such a problem could be overcome by averaging WACP measurements during the spot Hessian estimation process. However, with:

- at least two seconds per raw WACP measurement
  - ↪ determined by spectrum analyzer sweep, resolution/video BW
- at least 5 raw WACP measurements per averaged WACP measurement
  - ↪ to adequately reduce variance
- at least $3n^2+3n$ averaged WACP measurements per Hessian matrix estimate
  - ↪ determined by the robust *hybridized* technique of Appendix B.2
- at least 10 spot Hessian matrix estimates per Hessian Interval matrix / region
  - ↪ to reduce sampling bias
- many regions initialized / branched over the lifetime of the algorithm

this averaging would lead to an excruciatingly slow bounding / convergence rate and hence be practically unviable even for small variable vectors.

Despite being resilient to poor scaling, this last ✗ renders the algorithm unsuitable for our predistortion application.

Genetic (Global)

✓ Global convergence is unaffected by the *stochastic* nature of this algorithm provided the following measures are taken:

1. An initially large number of chromosomes χ are presented for selection into the population pool.
2. Fresh chromosomes are introduced into the mating pool of each evolutionary refinement cycle.
3. Evolutionary refinement continues until a small number ϑ of distinct chromosome concentrations form.

Measures 1 and 2 ensure appropriate seeding whilst measure 3 ensures non-premature exiting of the algorithm in accordance with CM 5 of Section 10.2.

✓ Since parent genes are treated independently during *reproduction*, pre-scaling *initial gene bounds* and *mutation variance* with respect to nonlinear order allows the algorithm to accommodate poor scaling. This is in accordance with counter-measure CM 2 of Section 10.2. As was the case for the Nelder-Mead Simplex algorithm, pre-scaling magnitude for each nonlinear order is set to be the variable increment size which induces a small but observable change in the output distortion spectrum.

✓ WACP randomness can be envisioned as chromosome fitness jitter. In the context of *selection* and *reproduction*, chromosome fitness jitter will introduce parent gene mutation and hence increase the effective variance of offspring mutation. By proactively reducing the standard *Normal* variance of offspring mutation by 5%, this additional parent mutation can be successfully accommodated. In the context of the *population pool* and *candidate solution space*, chromosome fitness jitter will mutate chromosome concentration regions. Since mutation is unbiased however, these concentration regions remain centered on candidate minima and the algorithm proceeds as normal. In both cases above, this gradient-free algorithm is seen to cope with WACP randomness without having to rely on WACP measurement averaging and is thus capable of large-scale optimization.

With its resilience to poor scaling and WACP randomness, this algorithm is considered highly robust and therefore well suited to our predistortion application.

11.4 Selected Optimization Algorithms

We conclude from this overall analysis that the local *Nelder-Mead Simplex* and global *Genetic* algorithms are the most suitable shortlisted algorithms for our predistortion application. Both of these algorithms are robust, gradient-free, objective measurement and comparison strategies. Their choice is thus in accordance with CM 6 of Section 10.2. Whilst all other shortlisted algorithms may have far superior search strategies (being gradient-based), their downfall is the excessive time spent estimating Gradient and Hessian characteristics of the WACP objective function.

Internal Parameter Settings

As discussed in Sections A.4 and A.5, the *Nelder-Mead Simplex* and *Genetic* algorithms possess several internal parameter settings. While hardware testing has revealed that algorithm performance is not overly sensitive to *common sense* parameter settings, the values outlined below are recommended as an initial guide.

For the *Nelder-Mead Simplex* algorithm (summarized in Figure A.6), three parameters require setting. The first is the simplex reset rate which controls stagnation and hence rate of convergence. A 40 iteration reset rate is used successfully in our performance testing. The second parameter that needs setting is d, the starting point increment of the initial simplex. As stated in Section A.4, this just needs to be set small compared to the dimension of the search space. We use a proportional value of 10 % in our performance testing. The final parameter that needs setting is the exit criterion Υ_0. Once again, this just needs to be set small and a value of $\Upsilon_0 = 10^{-2}$ has proven very reliable in our testing.

For the *Genetic* algorithm (summarized in Figure A.10), five parameters exist; although only two of these are considered *primary*, requiring upfront setting. The remaining three parameters, considered *secondary*, are derived accordingly. The first of the *primary* parameters is the working Population Pool size ξ, which we set to 50 chromosomes in our performance testing. The second of the *primary* parameters is the number of chromosome concentration regions ϑ at which evolutionary refinement is halted. As stated in Section A.5, this just needs to be set small and we find a value of $\vartheta = 4$ is very reliable.

The three *secondary* parameters are the Mating Pool size κ, Offspring Pool size ϱ and initial Population Pool size χ. These are logically derived from the *primary* working Population Pool size ξ, to implement the *Selection*, *Reproduction* and *Seeding* functions respectively. With our earlier setting of $\xi = 50$, values of $\kappa = 10$, $\varrho = 50$ and $\chi = 200$ are derived accordingly.

11.5 Refined Optimization Schedules

With the *Nelder-Mead Simplex* and *Genetic* algorithms now determined to be the local and global algorithms of choice for our predistortion application, Tables 11.2 and 11.3, with refined third column optimizers, hereby represent the finalized *Initial Setting* and *On-Air Adaption* schedules.

Optimization	$\boldsymbol{h_s}$	**Optimizer**	**WACP Weighting & Taper**
1a.	$\boldsymbol{p}_{3R}$	Genetic	Quadratic, $W_E = 10^5$
1b.	$\boldsymbol{p}_{3I}$		
2a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		Quadratic, $W_E = 10^4$
2b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		
3a.	$\boldsymbol{p}_{3R}$	Nelder-Mead Simplex	Quadratic, $W_E = 10^3$
3b.	$\boldsymbol{p}_{3I}$		
4a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		Quadratic, $W_E = 10^2$
4b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		

Table 11.2: Finalized *Initial Setting* optimization schedule

Optimization	$\boldsymbol{h_s}$	**Optimizer**	**WACP Weighting**
1a.	$\boldsymbol{p}_{3R}$	Nelder-Mead Simplex	Quadratic, $W_E = 10^2$
1b.	$\boldsymbol{p}_{3I}$		
2a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		
2b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		

Repeat indefinitely for constant tracking

Table 11.3: Finalized *On-Air Adaption* optimization schedule

Chapter 12

Pruning The Predistorter Volterra Kernel

In this chapter, we cover the final implementation aspects of the predistortion filter architecture, specifically Volterra Series pruning and associated memory estimation. This subsequently allows us to refine the optimization vector space $\boldsymbol{h}$ and hence gain a practical understanding of optimization loading.

Currently, our predistortion filter is modeled by the *Baseband* Volterra Series with no linear distortion and maximum 9^{th} order nonlinearity. This model was derived chronologically as follows:

1. In Section 5.2, the power amplifier was modeled as the *pure* Volterra Series in the RF transmitter model. Upon converting this RF transmitter model to its baseband equivalent, the power amplifier was found to take on the variant *Baseband* Volterra Series architecture characterized by odd ordered, complex kernels and conjugated product terms.

2. In Section 5.3, it was postulated that the predistortion filter with the greatest potential for canceling the nonlinear memory effects of the amplifier would be one that shares the same general architecture as the amplifier. Based on this thinking, the predistortion filter was modeled as the discrete-time *Baseband* Volterra Series.

3. In Section 6.2, the predistortion filter's first order operator was set to be transparent (unit impulse kernel) since digital predistortion, as standard practice, does not attempt to compensate for an amplifier's linear distortion.

4. In Chapter 8, a trade off analysis between linearization performance and computational complexity saw the the maximum order of predistorter nonlinearity chosen to be 9.

In line with the original operator notation of Chapter 5, the predistortion filter architecture is thus represented as:

$$\boldsymbol{P}\big[\, s[k] \,\big] = \sum_{\text{odd } m=1}^{\infty} \boldsymbol{P}_m\big[\, s[k] \,\big]$$

where

$$\boldsymbol{P}_m\big[\, s[k] \,\big] = \begin{cases} s[k] \quad \text{for } m = 1 \\ \\ \displaystyle\sum_{i_1=0}^{\infty} \cdots \sum_{i_m=0}^{\infty} p_m[i_1, \ldots, i_m]\, s[k-i_1] \cdots s[k-i_{\lceil \frac{m}{2} \rceil}]\, s^*[k-i_{\lceil \frac{m}{2} \rceil+1}] \cdots s^*[k-i_m] \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for } 3 \le m \le 9 \\ \\ 0 \quad \text{for } m \ge 11 \end{cases}$$

In this chapter, we choose a more compact representation for this predistortion filter, derived by absorbing all three m^{th} order conditions into a single equation and grouping *unconjugated / conjugated* input product terms:

$$\boldsymbol{P}\big[\, s[k] \,\big] = \; s[k] \; + \sum_{a=1}^{4} \left[\sum_{i_1=0}^{\infty} \cdots \sum_{i_{2a+1}=0}^{\infty} \left(p_{2a+1}[i_1, \ldots, i_{2a+1}] \prod_{x=1}^{a+1} s[k-i_x] \prod_{y=a+2}^{2a+1} s^*[k-i_y] \right) \right] \tag{12.1}$$

As discussed in Chapter 5, the large kernel of this generally applicable architecture requires pruning prior to implementation. Pruning is the process of stripping away predicted non-dominant kernel coefficients. Since the amplifier memory effects we are attempting to cancel are specific in origin and decline with time [299], most kernel coefficients of (12.1) will actually be non-dominant and hence have little effect in the linearization process. Pruning will thus dramatically reduce the size of our predistortion filter kernel and hence optimization vector space $\boldsymbol{h}$, without adversely affecting linearization potential. In accordance with counter-measure CM 1-C of Section 10.2, a smaller optimization vector space ultimately increases our optimization convergence reliability.

12.1 The Pruning Strategy

The pruning strategy we ultimately derive must have origins within the power amplifier modeling community. This is in accordance with our original postulation that *the predistortion filter with the greatest potential for canceling the nonlinear memory effects of the amplifier, is one that shares the same architecture as the amplifier.*

The power amplifier modeling community propose several pruning strategies including *Physical Knowledge*, *Near Diagonality Restriction*, *Dynamic Deviation Reduction* and *Volterra Behavioral Wideband*. As outlined below, each of these strategies use different mechanisms for predicting the dominant kernel coefficients. For a complete treatise of each strategy, the reader is directed to the references provided:

Physical Knowledge [305] - In this strategy, physical behaviors of a broad range of real amplifiers are summarized and abstracted to form a functional block model possessing feedback. Coefficients of this model are then regrouped and generalized to form equivalent Volterra kernels.

Near Diagonality Restriction [299, 306] - This strategy is an extension of the well known *memory polynomial* model [138, 147] with near-diagonal memory terms being retained in addition to those residing on the diagonal. Near-diagonal terms are restricted to offset columns of the predefined V-Vector input structure whose dimensions are determined by memory.

Dynamic Deviation Reduction [302, 304] - In this strategy, static nonlinearities and low order dynamics are identified to be the dominant sources of distortion in real power amplifiers. Kernel coefficients are subsequently differentiated according to their *dynamic order* and retained only if low ordered.

Volterra Behavioral Wideband [45] - Under the assumption of wideband operation, this strategy identifies dominant frequency-domain interactions existing between dependent control sources and ports of a general nonlinear network. These interactions, represented by multi-dimensional nonlinear transfer functions, are subsequently transformed to the time-domain via Fourier Inverses, to obtain the dominant Volterra kernel coefficients.

Of all these pruning strategies, *Volterra Behavioral Wideband* is considered the most suitable for wideband applications based on its operational bandwidth assumptions and frequency-domain development. For this reason, it is chosen as our basis strategy and applied to (12.1) to give:

$$\boldsymbol{P}\big[\, s[k] \,\big] = \; s[k] \; + \; \sum_{a=1}^{4} \left[\sum_{i_1=0}^{M} \cdots \sum_{i_a=0}^{M} \left(p_{2a+1}[i_1, \ldots, i_a]\, s[k] \prod_{x=1}^{a} \big| s[k-i_x] \big|^2 \right) \right] \quad (12.2)$$

It can be seen from (12.2) that the 3^{rd} order kernel ($a = 1$) now grows linearly with respect to finite memory length M and is hence a practically manageable size. However, the same cannot be said for higher order kernels which continue to grow exponentially. At this point we have three options:

1. Stick with this wideband pruning strategy, accepting the risk that optimization convergence reliability could be jeopardized by high order kernel size.

2. Choose a more aggressive narrower-band pruning strategy but jeopardize overall linearization performance due to lack of model fidelity.

3. Apply additional higher order pruning.

Since the first two options aren't suitable from a performance perspective, we are forced in the direction of additional higher order pruning and hence an overall *hybridized* pruning strategy.

For this additional higher order pruning, we turn to the *Dynamic Deviation Reduction* strategy and its fundamental principle that *high dynamic order terms contribute very little to model fidelity and hence can be removed to reduce kernel size*. The *dynamic order* of a term in the baseband Volterra Series refers to the number of delayed inputs forming that term. This should not be confused with the *nonlinear order* of a term which refers to the total number of inputs forming that term. For the *Volterra Behavioral Wideband* pruning of (12.2), the *dynamic order* of a $(2a+1)^{\text{th}}$ *nonlinear order* term ranges from 0 to $2a$. For example the arbitrary term:

$$p_7[3,2,5]\ s[k]\ \big|s[k-3]\big|^2\ \big|s[k-2]\big|^2\ \big|s[k-5]\big|^2 \tag{12.3}$$

has a nonlinear order of 7 (a = 3) and a dynamic order of 6, whilst the arbitrary term:

$$p_7[0,2,5]\ s[k]\ \big|s[k]\big|^2\ \big|s[k-2]\big|^2\ \big|s[k-5]\big|^2 \tag{12.4}$$

has a nonlinear order of 7 (a = 3) but a dynamic order of 4.

Since the 3rd order kernel of (12.2) comprises terms up to 2nd dynamic order, it is reasonable to restrict higher order kernels to 2nd dynamic order also. Applying this *Dynamic Deviation Reduction* to (12.2) then gives:

$$\boldsymbol{P}\big[\ s[k]\ \big] = \ s[k]\ + \sum_{a=1}^{4}\left[\sum_{i=0}^{M} p_{2a+1}[i]\, s[k]\,\big|s[k]\big|^{2(a-1)}\,\big|s[k-i]\big|^2\right] \tag{12.5}$$

It can be seen from (12.5) that all kernels now grow linearly with respect to memory length M and are hence a practically manageable size. While this desirable outcome marks the end of pruning from the perspective of *electrical amplifier matching*, one further *implementation related* pruning opportunity exists as follows.

The input signal to the digital predistortion filter must be oversampled (Inphase and Quadrature) by at least the highest predistortion filter nonlinearity in order to prevent spectral regrowth aliasing. This heavy oversampling leads to an input signal

with a very narrow discrete-time spectral bandwidth $B_d = B_c/F_s$. Here, B_c is the signal's continuous-time spectral bandwidth and F_s is the sampling frequency. As a result, the change between adjacent input signal samples can be considered very small to the point where groups of R input samples (R being small) can be assumed equal. With this assumption, (12.5) can be refined with *R-sample delay increments* instead of *single-sample delay increments* without loss in fidelity as follows:

$$\boldsymbol{P}\big[\, s[k] \,\big] = \; s[k] \; + \sum_{a=1}^{4} \left[\sum_{i=0}^{\lceil \frac{M+1}{R} \rceil - 1} \tilde{p}_{2a+1}[i]\, s[k] \big| s[k] \big|^{2(a-1)} \big| s[k - Ri] \big|^{2} \right] \tag{12.6}$$

In this refinement, $\lceil \cdot \rceil$ represents the ceiling operator and $\tilde{}$ denotes kernel transformation. The precise form of this transformation is considered irrelevant however since (12.6) kernel estimation is performed independently of (12.5).

These larger R-sample delay increments not only lead to a further (approximate) R-fold pruning of the predistortion filter kernel but also make the architecture independent of hardware sample rate implementation. The value of R must be estimated from the input signal's discrete-time bandwidth with a smaller discrete-time bandwidth allowing a greater value of R and vice versa. Discrete-time bandwidths and corresponding values of R, as used in the experimental analysis of this research, are presented in Table 12.1.

Signal Modulation	$\boldsymbol{B_c}$ **(MHz)**	$\boldsymbol{F_s}$ **(MHz)**	$\boldsymbol{B_d = B_c/F_s}$ **(cycles / sample)**	$\boldsymbol{R}$ **(samples)**
DAB	1.537	65.536	0.023	5
WCDMA	4.096	92.160	0.044	3
DVB-T	6.656	64.000	0.104	2

Table 12.1: R-sample delay increments

It is noted here that the sampling rate of the DVB-T signal is lower than that of the DAB and WCDMA signals despite exhibiting a greater continuous-time bandwidth. This is because the next power of two IFFT would bring the sampling rate to greater than what the implementation hardware is capable. The current sampling rate is satisfactory however given it oversamples the DVB-T signal by at least the highest predistortion filter nonlinearity and therefore avoids spectral regrowth aliasing.

In all cases, a conservative estimate of R is advisable. Over-estimating R in

order to gain extra kernel pruning will only invalidate the underlying assumption that *groups of R input samples are equal* and hence lead to degraded linearization performance. It is also noted that although the predistortion filter has been refined to operate with internal R-sample delay increments, it is still clocked at the oversampling rate to avoid spectral regrowth aliasing.

The novel, triple pruned baseband Volterra Series (12.6) represents the final digital predistortion filter architecture. Quite simply, any further pruning will remove dominant kernel coefficients and hence jeopardize linearization performance. This predistortion filter architecture is implemented in software on the laboratory transmitter testbed, specifically via the object class *Predistorter Volterra* and its member function *filter()*. Corresponding class-declaration and function-definition source code resides in project files *Predistorter Volterra.h* and *Predistorter Volterra Templates.cpp* respectively. Both files are located within folder *Software\Cpp* on the accompanying DVD.

Based on its derivation, we call the novel hybridized pruning strategy *Volterra Behavioral Wideband with Reduced Dynamic Order and R-Sample Delay Increments*. For clarity, this final digital predistortion filter architecture is expanded in (12.7) in terms of its individual nonlinear operators. In this representation, kernel accentuation is dropped for the sake of notation simplicity:

$$\begin{aligned}
\boldsymbol{P}\big[\, s[k] \,\big] \;=\; & s[k] \\
& + \sum_{i=0}^{\lceil \frac{M+1}{R} \rceil - 1} p_3[i]\; s[k]\; \big|s[k-Ri]\big|^2 \\
& + \sum_{i=0}^{\lceil \frac{M+1}{R} \rceil - 1} p_5[i]\; s[k]\; \big|s[k]\big|^2 \big|s[k-Ri]\big|^2 \\
& + \sum_{i=0}^{\lceil \frac{M+1}{R} \rceil - 1} p_7[i]\; s[k]\; \big|s[k]\big|^4 \big|s[k-Ri]\big|^2 \\
& + \sum_{i=0}^{\lceil \frac{M+1}{R} \rceil - 1} p_9[i]\; s[k]\; \big|s[k]\big|^6 \big|s[k-Ri]\big|^2
\end{aligned} \tag{12.7}$$

In the next section, we set out to estimate the predistortion filter's finite memory length M; the final variable in (12.7) yet to be quantified.

12.2 Memory Estimation

The most intuitive approach to estimating predistortion filter memory is to first estimate transmitter memory, via test signal injection and vector network analysis / correlation [166,202,279], and then translate this estimate back across to the predistortion filter. As signal modulation bandwidth grows however, this system identi-

fication approach becomes increasingly inaccurate, specifically under-estimating [80, 239, 272]. Hence for applications such as DAB, WCDMA and DVB-T, we propose a more reliable experimental procedure for estimating predistortion filter memory. This procedure is based on probing the transmitter with memory specific distortion created by the predistortion filter and observing its effect on the signature of the output adjacent channel power spectrum. *Signature* here refers to the *shape* of the spectrum. The idea is simple; if a memory specific probing shows potential in reducing spectral power evenly across the adjacent channel band, then that memory component is considered present.

It is important to understand that in this procedure, the predistortion filter is kept in place but isn't operated in the normal sense to linearize the transmitter. Rather, it is operated specifically to probe the transmitter with memory specific distortion. This involves:

1. setting its internal R-sample delay increment to unity. As will become evident later, this allows a finer memory estimate to be achieved.

2. setting a single 3^{rd} order kernel coefficient $p_3[i]$ to a nonzero value whilst keeping all other kernel coefficients zero. Here, the choice of i ultimately determines the memory associated with the probing distortion. In this respect, the predistortion filter reduces to:

$$\boldsymbol{P}\big[\, s[k] \,\big] = s[k] + p_3[i]\, s[k] \big| s[k-i] \big|^2 \tag{12.8}$$

   The nonzero 3^{rd} order kernel coefficient $p_3[i]$ can be chosen as real or complex although experience has shown real to be sufficient. Its sign and magnitude are chosen with the intent of reducing the adjacent channel power spectrum by an observable amount ($\approx$2 dB).

Since system memory is responsible for frequency dependent behavior [52, 53], probing the transmitter with memory specific distortion i will have the following qualitative effects:

- When i is set less than the required memory, the probing distortion can be expected to add destructively (assuming correct sign of $p_3[i]$) with existing transmitter distortion and the resulting power spectrum will generally reduce evenly across the adjacent channel band. In this sense, the signature of the adjacent channel power spectrum does not change.

- When i is set greater than the required memory, the probing distortion can be expected to have no memory matched transmitter distortion and as a result the predistortion filter will effectively introduce distortion which has a different

adjacent channel spectral signature to any distortion which currently exists. It follows that the resulting adjacent channel power spectrum will not only change magnitude, but more importantly change its spectral signature.

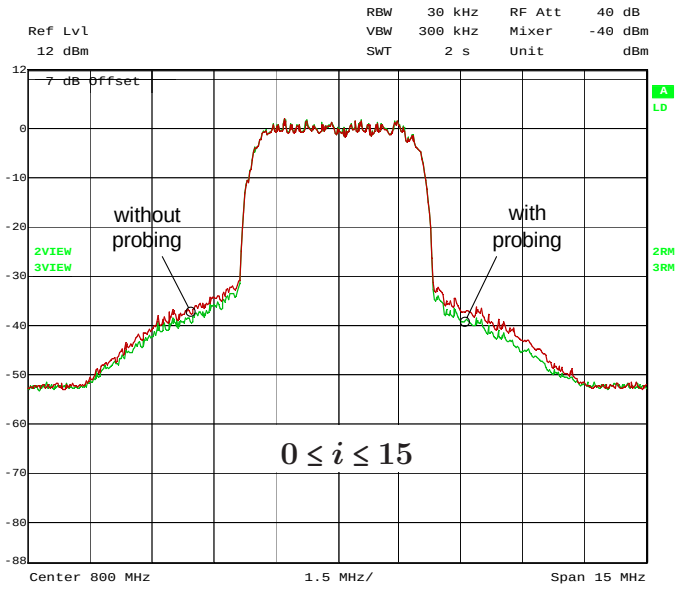
It logically follows that if a sweep of i from unity upwards is performed, a change in the signature of the adjacent channel power spectrum will be observed when i reaches the required memory. It is this signature changing value of i which we use to estimate M.

This concept of sweep-probing the transmitter with memory specific distortion and looking for changes in the signature of the adjacent channel power spectrum is demonstrated in Figures 12.1 / 12.2 for the laboratory transmitter testbed and WCDMA / DVB-T signal modulation. In the following analysis, we focus on the WCDMA case alone, purely to reduce the amount of subfigure referencing, however the exact same conclusions can be drawn from the DVB-T case.

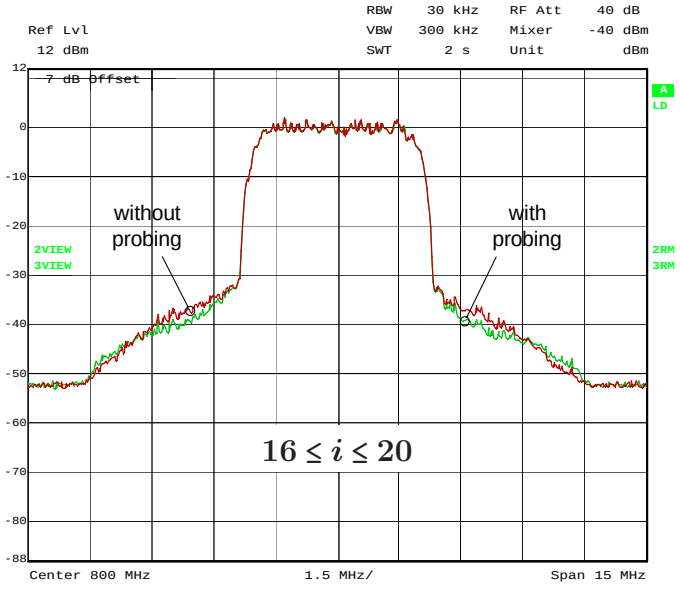
Subfigure 12.1a is representative of the output power spectrum after probing with $0 \leq i \leq 15$, Subfigure 12.1b is representative of the output power spectrum after probing with $16 \leq i \leq 20$ and Subfigure 12.1c is representative of the output power spectrum after probing with $21 \leq i \leq 30$. It is noted that ranges of i are presented here, instead of individual values, to limit the number of plots.

It is observed that for $0 \leq i \leq 15$ (Subfigure 12.1a), the signature of the probed spectrum remains similar to that of the unprobed spectrum, remembering that *signature* here refers to the *shape* of the spectrum, not its magnitude. The observed change in magnitude is due to the probing distortion adding destructively with existing transmitter distortion.

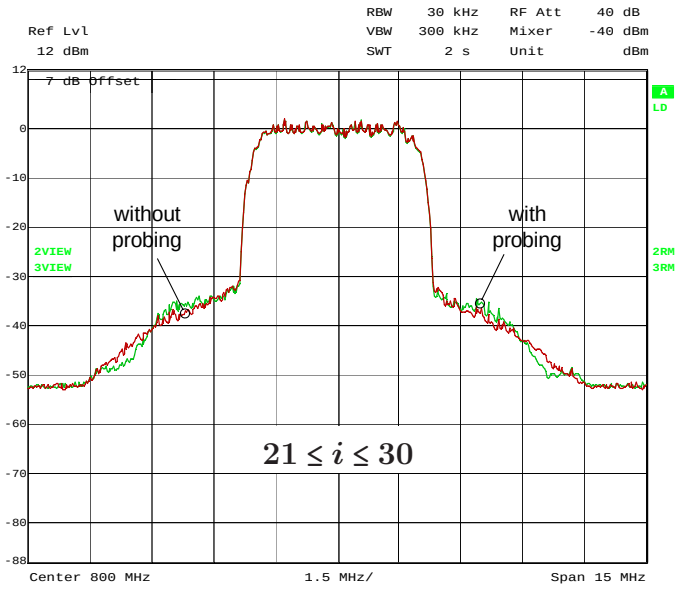
As i increases however, in this case for $16 \leq i \leq 20$ (Subfigure 12.1b), the signature of the probed spectrum begins to change with a slight ripple forming across the adjacent channel. This range of i marks the point when the probing distortion runs out of memory matched transmitter distortion and therefore the probing predistortion filter is effectively introducing distortion which has a different spectral signature to any transmitter distortion which currently exists. As demonstrated here in practice, the change in spectral signature is a continuous process occurring over several samples, in this case 5 ($16 \leq i \leq 20$). In this sense, ambiguity can arise in terms of whether to choose the first or last of these samples to represent predistortion filter memory M. However, it is always advisable to over-estimate rather than under-estimate memory for two reasons. Firstly, under-estimation is another form of Volterra Series pruning which leads to degraded linearization performance and secondly, as discussed in Section 9.2, the weighting of the WACP objective provides added robustness against redundant memory components. Based on this reasoning, predistortion filter memory is estimated here as $M = 20$.



(a)

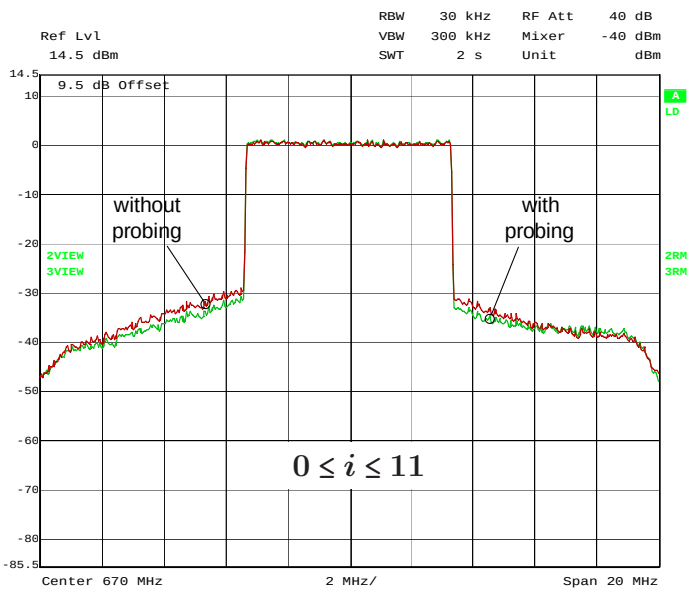


(b)

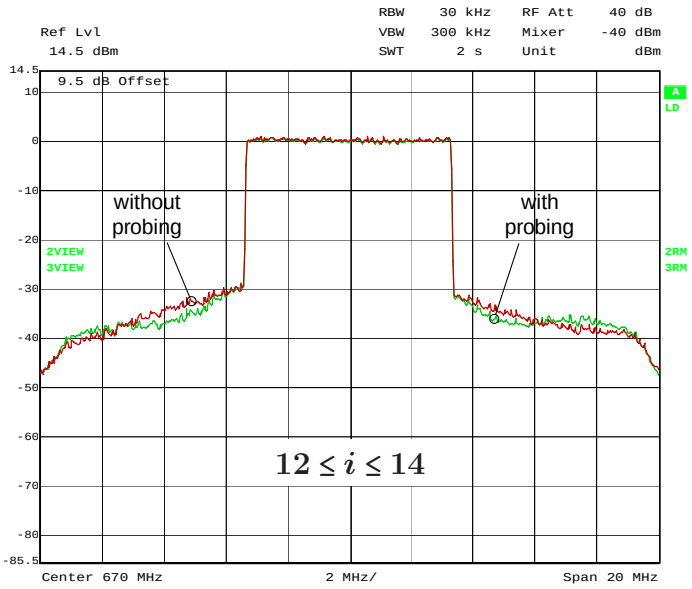


(c)

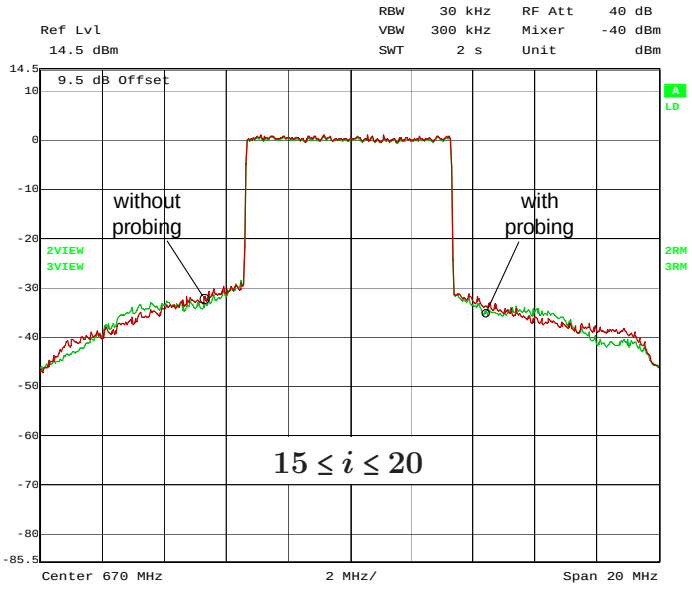
Figure 12.1: Output power spectrum of transmitter testbed (WCDMA signal modulation) with probing representative of (a) $0 \leq i \leq 15$ (b) $16 \leq i \leq 20$ (c) $21 \leq i \leq 30$



(a)



(b)



(c)

Figure 12.2: Output power spectrum of transmitter testbed (DVB-T signal modulation) with probing representative of (a) $0 \leq i \leq 11$ (b) $12 \leq i \leq 14$ (c) $15 \leq i \leq 20$

For $i > M$, in this case $21 \leq i \leq 30$ (Subfigure 12.1c), the ripple in the signature of the probed spectrum is observed to compress and slide up towards the transmit channel, hence making room for an additional ripple in the outer adjacent channel region (compared to Subfigure 12.1b). This pronounced rippling effect, and hence frequency dependent behavior, continues for $i > 30$.

Since predistortion filter memory M is assumed independent of nonlinear order in (12.7), we are in fact free to probe the transmitter with any order of nonlinear distortion via (12.8). 3^{rd} order is specifically chosen however since 3^{rd} order transmitter nonlinearity is dominant and hence its power spectrum, and any change in its power spectrum caused by the probing predistortion filter, is completely observable. This is not the case for higher orders.

Compared to the traditional approach of estimating transmitter memory and then translating this estimate back to predistortion filter memory, the proposed experimental approach is considered simpler because it avoids the need for specific test signal injection and vector network analysis / correlation. It is also considered more direct and accurate because it estimates memory directly with the predistortion filter in place, thus matching the predistortion filter architecture used. That is, the estimate is based on exactly what the predistortion filter and transmitter would experience in practice during optimization, thus resulting in a more accurate estimate.

Table 12.2 presents the memory estimates M obtained when this procedure is performed on the transmitter testbed for all target signal modulations. Here, T_s represents discrete-time sampling period whilst M_c represents continuous-time equivalent memory. As can be seen from the last column, M_c is virtually constant across the different signal modulations, thus providing confirmation of the estimates M. These memory estimates are also consistent with independent general estimates presented in the literature [44, 45, 296, 303].

Signal Modulation	$\boldsymbol{T_s}$ **(nsec)**	$\boldsymbol{M}$ **(samples)**	$\boldsymbol{M_c = MT_s}$ **(nsec)**
DAB	15.259	14	213.623
WCDMA	10.851	20	217.014
DVB-T	15.625	14	218.750

Table 12.2: Predistortion filter memory estimates

12.3 Refined Optimization Vector Space

The optimization vector space $\boldsymbol{h}$ was initially defined in Section 9.1 in terms of the predistortion filter's *unpruned* Volterra kernel. Specific reference is made to (9.1) and (9.2) on page 73. With the predistortion filter now pruned according to (12.7) however, this definition can be refined as:

$$\boldsymbol{h} = \left[\boldsymbol{p}_3 \mid \boldsymbol{p}_5 \mid \boldsymbol{p}_7 \mid \boldsymbol{p}_9 \right] \tag{12.9}$$

where for $m = 3, 5, 7, 9$:

$$\boldsymbol{p}_m = \left\{ p_m[i] \quad \text{for} \quad i = 0 \rightarrow \left\lceil \frac{M+1}{R} \right\rceil - 1 \right\} \tag{12.10}$$

Here, $p_m[i]$ represents the pruned m^{th} order Volterra kernel of (12.7) and $\lceil \cdot \rceil$ represents the ceiling operator. With the vector space now finite in dimension, its overall size S can be formulated as:

$$S = 4 \left\lceil \frac{M+1}{R} \right\rceil \text{ complex variables} \tag{12.11}$$

Directly related to S, though slightly more insightful in terms of optimization convergence reliability, the maximum optimization load L can also be formulated as:

$$L = 3 \left\lceil \frac{M+1}{R} \right\rceil \text{ real variables} \tag{12.12}$$

L represents the size of the largest optimization subsets defined within the *Initial Setting* and *On-Air Adaption* schedules (page 100), specifically $\left[\boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \right]$ and $\left[\boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \right]$. Expectedly, L grows with predistorter memory length M and decreases with the coarseness of the R-sample delay increment. With the values of R and M already quantified in Tables 12.1 and 12.2 respectively, L for each target application can be quantified via simple (12.12) substitution, as summarized in Table 12.3.

Signal Modulation	B_c **(MHz)**	R **(samples)**	M **(samples)**	L **(real variables)**
DAB	1.537	5	14	9
WCDMA	4.096	3	20	21
DVB-T	6.656	2	14	24

Table 12.3: Maximum optimization load L for each target application

It can be seen from this table that maximum optimization load L increases with signal modulation bandwidth B_c, reaching a maximum of 24 real variables for DVB-T. Providing Section 11.3 and 11.4 guidelines are adhered to, this figure is easily accommodated by the robust Nelder-Mead Simplex and Genetic algorithms and therefore any concerns regarding optimizer loading and convergence reliability can be laid to rest.

Before leaving this chapter, it is important to reiterate the role of pruning in the context of the entire algorithm. As initially outlined in the introduction of Chapter 5, we chose from the very outset to develop this predistortion algorithm around an unpruned Volterra Series model in order to establish generality and widespread applicability. Only at the very end would we apply pruning to match the algorithm to the target modulation standard. Based on this design principle, the pruning strategy can be considered independent of the core underlying predistortion algorithm and hence completely interchangeable if the need ever arises in the future. For instance, if an alternative (yet to be developed) pruning strategy is identified to be more suited to the power amplifier hardware or modulation standards of the future, it is completely feasible to replace the hybrid pruning strategy of this chapter without affecting the operation of the core predistortion algorithm. This front-end interchangeability leads to an algorithm with an indefinite application life.

As alluded to above, this chapter on predistortion filter kernel pruning marks the end of all algorithm development. In the following chapters we turn attention to algorithm performance analysis.

Chapter 13

Performance Baseline

In this chapter, we review the quantitative performance of current wideband predistortion systems in order to derive a performance baseline for our technique. This baseline, to be used in the next chapter's performance testing, will represent the minimum level of performance our technique must achieve if it is to be considered state-of-the-art.

Predistortion performance is specifically quantified in the frequency-domain by *ACD Shoulder Height* and *ACD Shoulder Attenuation*, as demonstrated in Figure 13.1. *ACD Shoulder Height* is a measure of how nonlinear the transmitter is driven prior to predistortion whilst *ACD Shoulder Attenuation* is a measure of distortion reduction. Since *ACD Shoulder Height* sets the difficulty of the linearization problem, the two quantities must be considered as a [Height / Attenuation] pair. For example, [-30 dBc / 15 dB] is considered better performance than [-40 dBc / 15 dB] since the former is a more difficult linearization problem.

Our performance review spans academic and industry sources including *Research Journals*, *Textbooks*, *Application Notes*, *Transmitter Manufacturers*, *Transmitter Operating Manuals* and *Transmission Service Providers*. From these sources, we only consider *wideband*, *hardware tested* and *adaptive* predistortion techniques. This criteria ensures the derived quantitative performance baseline is an *accurate* and *unbiased* representation of predistortion state-of-the-art.

For clarification, we do not consider *narrowband*, *model tested* or *nonadaptive* techniques for the following reasons. *Narrowband* techniques demonstrate biased performance since they do not need to compensate for complex dynamic amplifier memory [189, 216, 222]. *Model tested* techniques also demonstrate biased performance since their computer models can, at best, only approximate a real wideband amplifier's complex dynamic memory [181, 209, 244]. *Nonadaptive* techniques are unable to maintain long-term spectral mask compliancy due to transmitter nonlin-

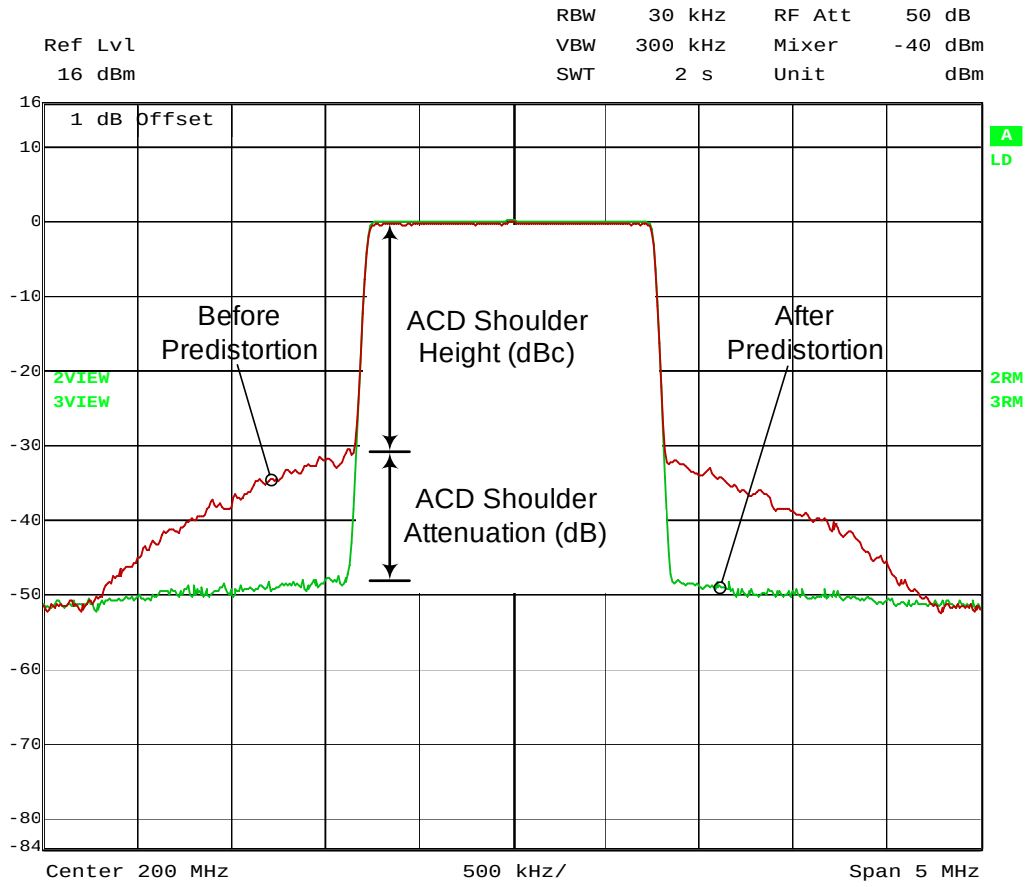


Figure 13.1: Adjacent Channel Distortion (ACD) Shoulder Height and Attenuation

ear drift [50, 107, 288]. As such they cannot be commercially deployed and hence do not represent state-of-the-art.

With the findings from each review source presented in the following sections, it will be seen that different practical circumstances make some sources more productive than others. In the end, that current performance identified to be best will subsequently represent our performance baseline.

13.1 Research Journals

As discussed in Section 2.2 of the literature review, *Self-Learning* is the most prevalent *wideband* predistortion strategy. Of the many papers written on *Self-Learning*, only a small minority present *hardware tested* performance results. These are summarized in Table 13.1. As can be seen from the third column, we were unable to identify hardware tested cases for DVB-T or DAB, despite all three target modulations being equally represented in the literature. It can only be assumed that cellular hardware is more readily available in academic circles.

Case	Paper	Signal Modulation	Shoulder Height (dBc)	Shoulder Atten. (dB)	Comments
1	[288]	1-carrier WCDMA	-43	15	Adaptive ✓
2	[96]	8 tones over 4 MHz	-20	15	Low CF Test Signal ✗
3	[186]	15 MHz CDMA	-40	16	Nonadaptive ✗
4	[55]	3-carrier WCDMA	-35	20	Nonadaptive ✗
5	[99]	3-carrier WCDMA	-45	10	Nonadaptive ✗
6	[164]	1-carrier WCDMA	-34	20	Nonadaptive ✗
		3-carrier WCDMA	-34	16	Nonadaptive ✗

Table 13.1: Wideband, hardware tested performance as presented in research papers

Of the six *wideband*, *hardware tested* cases identified in Table 13.1, only **Case 1** can be considered relevant in terms of deriving a quantitative performance baseline for our technique. Here, the predistorter is *adaptive* and exhibits [-43 dBc / 15 dB] performance. The remaining cases are ruled out due to inappropriate test signaling or inability to adapt, as discussed below:

Case 2 - an eight-tone test signal is used to *simulate* WCDMA. Despite possessing similar bandwidths, the test signal's Crest Factor is approximately 3 dB lower than that of *real* WCDMA and hence doesn't truly replicate the high Crest Factor demands placed on current wideband communication systems. To be precise, the Crest Factor of this eight-tone test signal is on par with previous generation 16 QAM linear signal modulation (please refer to Table 1.1's Crest Factor comparisons on Page 7). This leads to an *over-relaxed* predistortion problem and generously biased test results; considering the very high -20 dB shoulder height. If this paper had used *real* WCDMA test signals, we would have considered its performance in setting our baseline.

Cases 3 & 4 - the algorithms are based on the standard *Indirect Learning* strategy. As discussed in Section 2.2.2 of the literature review, this strategy can only perform adaption by turning off the predistortion filter whilst on-air. This is specifically to allow fresh transmitter data to flow through to the output and be collected for postdistortion linear regression. Turning off the predistorter whilst on-air inevitably leads to unlinearized, high distortion transmission and

hence violation of legal spectral masks and heavy financial fines from regulators. Transmitter operators are not willing to be fined and hence such algorithms are considered commercially unviable. If the algorithms were seamlessly adaptive, we would have considered their performance in setting our baseline.

Cases 5 & 6 - the algorithms are based on a variation of the standard *Indirect Learning* strategy. Whilst this variation sees postdistortion parameters computed via a combination of linear regression and model inverses, instead of purely the former, the translation process is identical to that of standard *Indirect Learning* and hence the strategy suffers the same consequences as **Cases 3 & 4** above. That is, adaption can only be performed by turning off the predistortion filter whilst on-air, leading to violation of legal spectral masks and hence commercial unviability. If the algorithms were seamlessly adaptive, we would have considered their performance in setting our baseline.

Like most readers will be, we are slightly surprised by the lack of *wideband* and *hardware tested* and *adaptive* performance data presented in research journals. It can only be assumed that the financial cost of transmission hardware and test instrumentation poses a significant constraint for such researchers. Despite this unexpected finding, the [-43 dBc / 15 dB] performance of **Case 1** is a good starting point in our quest to derive a quantitative performance baseline for our technique.

13.2 Textbooks

As a whole, cellular and broadcast textbooks are more oriented towards theoretical concepts than applied performance. We did however manage to find one textbook which was more application related and touched on the subject. This book, *Digital Video and Audio Broadcasting Technology: A Practical Engineering Guide* [72], written by Walter Fischer and sponsored by Rohde & Schwarz, states that [-30 dBc / 10 dB] adaptive performances are achieved in practice for DVB-T. With the author having many years of experience in the industry, we consider this figure to be highly reliable.

For consistency, it is worthwhile noting that the [-43 dBc / 15 dB] performance identified in the previous Research Journal section *logically trends* with this newly identified [-30 dBc / 10 dB] performance. That is, the former less difficult linearization problem (lower ACD shoulder height) should, and indeed does, result in greater ACD shoulder attenuation. This builds confidence in our initial findings.

13.3 Application Notes

Despite many application notes being written on DVB-T and WCDMA, with focused sections on spectral masks and ACD shoulder measurement, only one was found that spoke directly about predistortion performance. The Rohde & Schwarz application note *Measurements on MPEG2 and DVB-T Signals - Part 3* [95], written by Sigmar Grunwald, states that adaptive performances around [-30 dBc / 10 dB] are practically achievable for DVB-T. This is consistent with the previous section's findings which also had links to Rohde & Schwarz.

13.4 Transmitter Manufacturers

With success in the previous two sections linked to a transmitter manufacturer, we felt further success may eventuate from direct contact with other global transmitter manufacturers, specifically NEC, Harris, Ericsson and Nokia Siemens. While the idea was sound, the result wasn't. On each occasion, we were advised that such information was Commercial-In-Confidence and it could not be disclosed. This response was consistent with their product marketing brochures lacking concrete performance figures. From this pursuit, it was obvious that Rohde & Schwarz was more transparent than others when it came to product performance.

13.5 Transmitter Operating Manuals

With no assistance coming from the transmitter manufacturers, we turned attention to their transmitter operating manuals, hoping that example performance data would be inter-dispersed with instruction sets. However, after acquiring manuals for the NEC DTU-31 5 KW UHF transmitter [194], NEC DTV-40 2.5 KW VHF transmitter [193] and Ericsson 2206 Radio Basestation [61], it soon became apparent that performance non-disclosure was still being silently enforced. Only basic instructions on how to *set* and *arm* the predistorter from the transmitter's front panel interface were presented.

13.6 Transmission Service Providers

Unsatisfied with the previous two outcomes and still determined to acquire applied performance data, we contacted local transmission service providers, asking if we could monitor the performance of their in-service systems. Two responded positively, these being Channel 7 and Broadcast Australia.

Channel 7

Channel 7 granted us access to their Yarrawonga transmitter.[1] Consisting of a Rohde & Schwarz SV702 exciter and VH6010A2 amplifier, this transmitter broadcasts 732.5 MHz DVB-T at 25 W EIRP. Accompanied by Channel 7 radio technician Ian Smart, the on-air amplifier output spectrum was measured with and without predistortion, via test port coupling. These measurements are presented below.

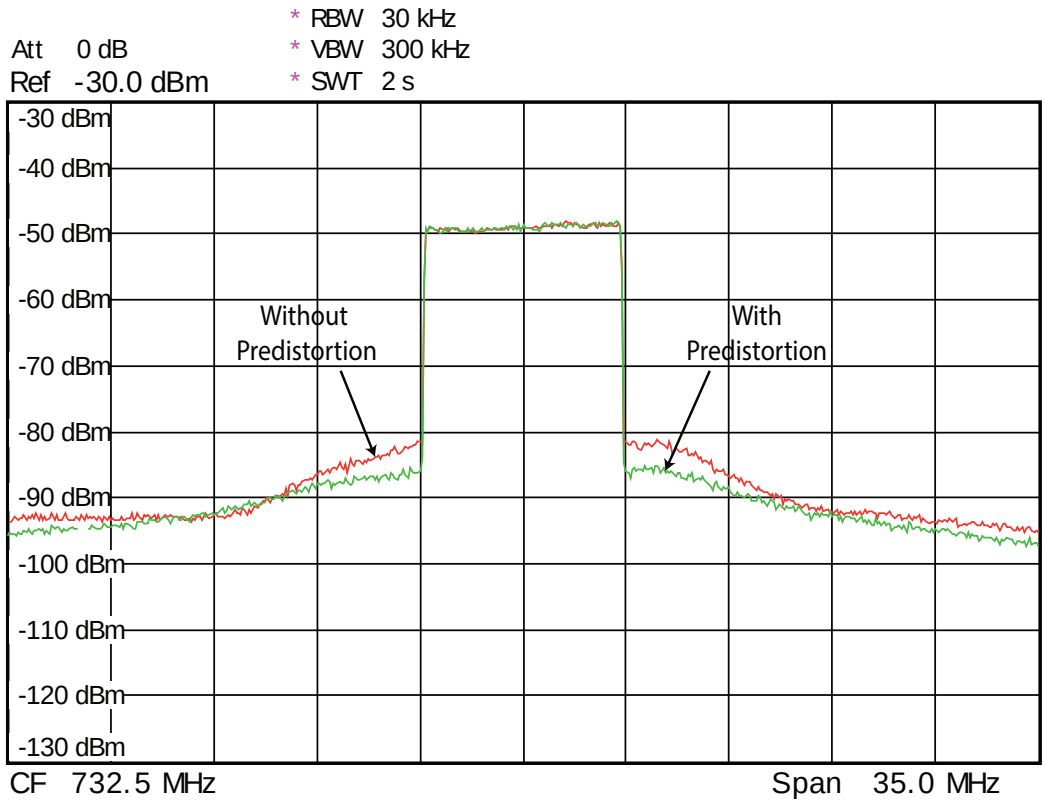


Figure 13.2: Channel 7 Yarrawonga amplifier output

With each trace labeled, we observe a performance of [-31 dBc / 5 dB]. Ian Smart further advised that this was a 1st generation digital transmission system and therefore predistortion performance was more than likely not state-of-the-art considering the processing refinements of later 2nd generation systems. It can only be assumed that the Rohde & Schwartz linked performance data obtained previously, with 5 dB better shoulder attenuation, applies to the latest 2nd generation systems.

[1] Yarrawonga transmission site services the North Ward area of Townsville.

Broadcast Australia

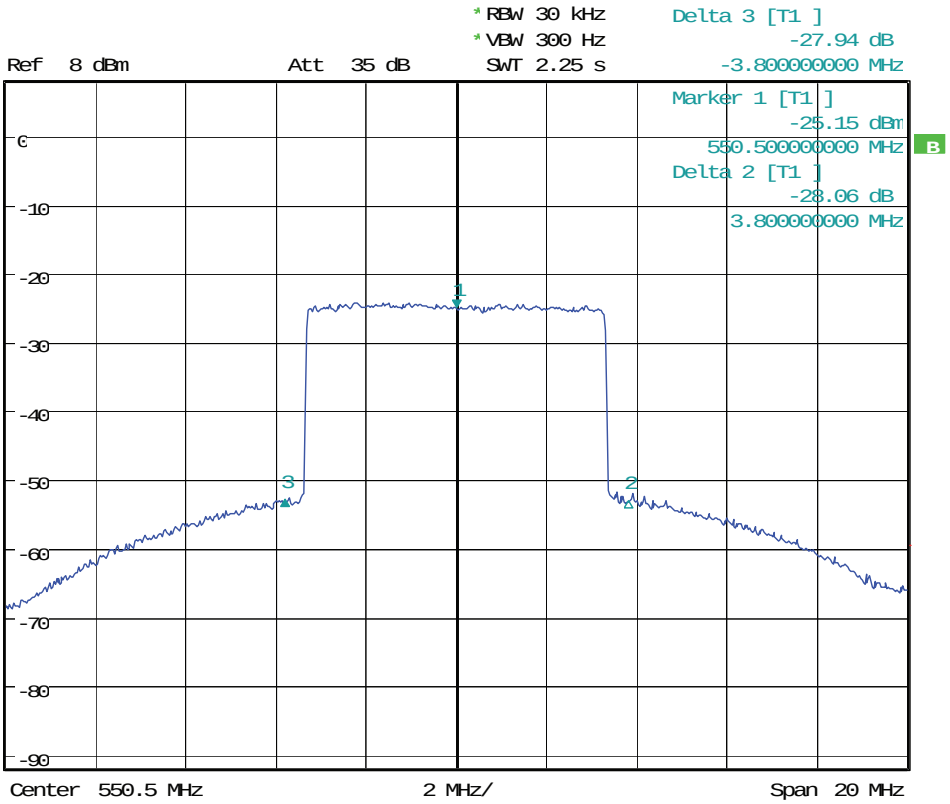
Whilst Broadcast Australia, couldn't grant us measurement access to any transmitters, they did provide us with pre-measured performance data for Mt Stuart's ABC and SBS transmitters.[2] The ABC transmitter, considered a 1st generation digital transmission system, is comprised of a Tandberg exciter and NEC power amplifier and broadcasts 550.5 MHz DVB-T at 200 KW EIRP. The SBS transmitter, considered a 2nd generation digital transmission system, is fully NEC and broadcasts 592.5 MHz DVB-T at the same EIRP.

As presented in Figures 13.3 and 13.4, the data provided by Broadcast Australia consists of the exciter and amplifier output spectra, in both cases with predistortion activated. No data for the predistortion deactivated case was available however. Raising this with Broadcast Australia, we were advised that since these transmitters are very high power, on-air deactivation of the predistorter requires regulatory consultation, due to high power spectral mask failure, and is therefore avoided except for very special circumstances. As in the lower power Yarrawonga case, having this extra data would have allowed us to define precise performance figures. However all is not lost.

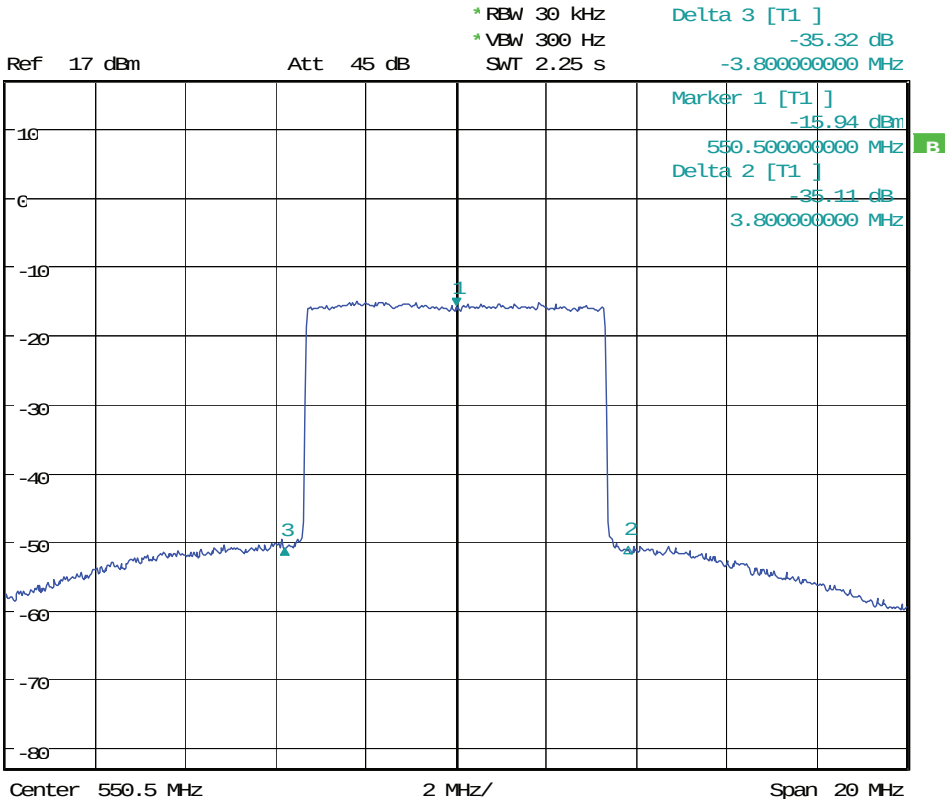
As discussed in Chapter 7 and specifically demonstrated in Figure 7.6 (page 69), the predistorter's ACD Shoulder Height must be greater than the amplifier's unlinearized ACD Shoulder Height if self-generated parasitic distortion is to be canceled. This means that the exciter ACD Shoulder Heights observed in Figures 13.3a and 13.4a represent an upper bound on the amplifiers' unlinearized ACD Shoulder Heights (the missing data) and therefore bounded performance can be defined as [< -28 dBc / < 7 dB] for the ABC transmitter and [< -26 dBc / < 12 dB] for the SBS transmitter. With a very conservative 3 dB bounding buffer estimate, derived from Figure 7.6's testbed measurements (page 69), these performances are estimated to be [-31 dBc / 4 dB] for ABC and [-29 dBc / 9 dB] for SBS.

Before leaving this section, it is worth noting that Lanecomm Communications, contracted to Kordia Solutions to support the North Queensland Vodafone network, were also very keen to assist with gathering performance data, specifically WCDMA. However, despite supervised access being granted to local Vodafone basestations, it was impossible to take on-air transmission measurements without interfering with network operation. Quite astonishingly, the Ericsson transmitters in use didn't possess external test ports or output coupling. Lanecomm also provided engineering contacts within Kordia and Telstra but none were able to provide assistance.

[2] Mt Stuart transmission site services the wider Townsville area.

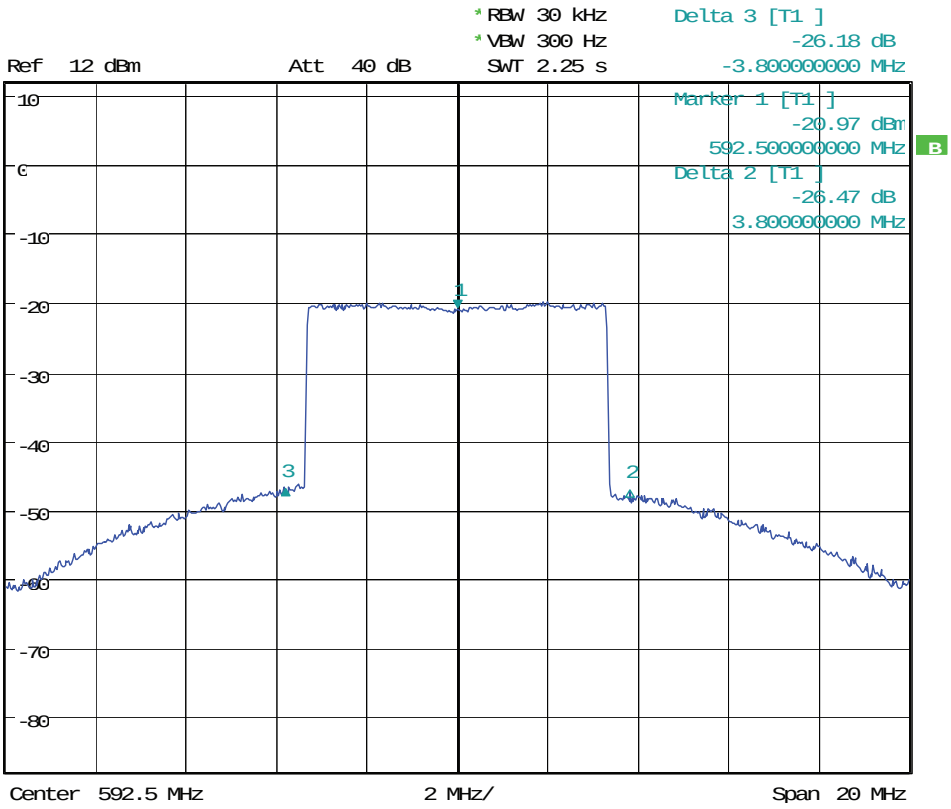


(a) Exciter output with predistortion

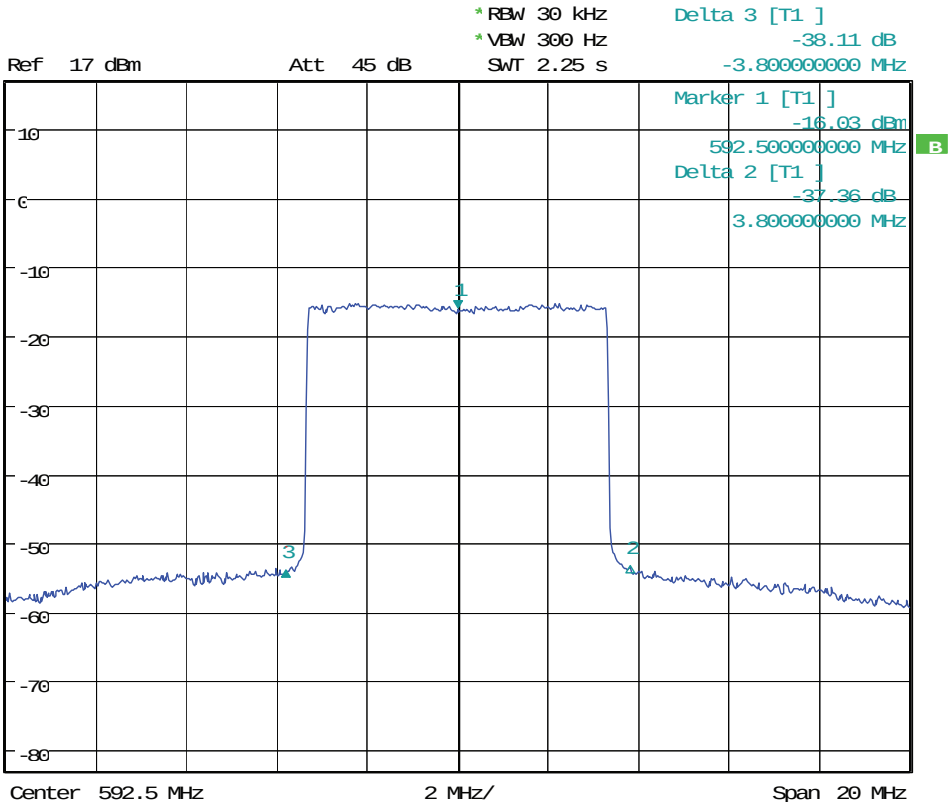


(b) Amplifier output with predistortion

Figure 13.3: ABC Mt Stuart transmission



(a) Exciter output with predistortion



(b) Amplifier output with predistortion

Figure 13.4: SBS Mt Stuart transmission

13.7 Derivation of Performance Baseline

Our review clearly demonstrates the nontriviality associated with obtaining *wideband* and *hardware tested* and *adaptive* predistortion performance data. For convenience, its findings are summarized below.

Source	Modulation	Performance	Comment
Research Journal [288]	WCDMA	[-43 dBc / 15 dB]	Lowest ACD Shoulder Height
Textbook [72]	DVB-T	[-30 dBc / 10 dB]	Linked To Rohde & Schwarz
Application Note [95]	DVB-T	[-30 dBc / 10 dB]	Linked To Rohde & Schwarz
Channel 7	DVB-T	[-31 dBc / 5 dB]	1st Generation Rohde & Schwarz
ABC	DVB-T	[-31 dBc / 4 dB]	1st Generation Tandberg / NEC
SBS	DVB-T	[-29 dBc / 9 dB]	2nd Generation NEC

Table 13.2: Summary of wideband, hardware tested, adaptive performance data

We immediately observe that the majority of reported performances are for DVB-T. It is stressed that this is a natural outcome of the review with all target modulations being considered equally in the review process. While more WCDMA and DAB performances would have been ideal for the sake of completeness, a performance baseline derived from current best DVB-T performance is more than sound. This is because DVB-T has the widest bandwidth, highest Crest Factor and highest amplifier drive level [3] of all three target modulations and is therefore considered the more difficult linearization problem. If our predistortion technique outperforms others based on DVB-T, it is guaranteed to outperform others based on WCDMA and DAB.

For all DVB-T cases, the ACD Shoulder Height is approximately -30 dBc. For the Channel 7 and ABC 1st generation systems, ACD Shoulder Attenuation is approximately 5 dB. For the SBS 2nd generation system, ACD Shoulder Attenuation increases to 9 dB. This 2nd generation NEC performance is consistent with the two other reports linked to Rohde & Schwarz in the literature.

With two different DVB-T transmitter manufacturers (NEC and Rohde & Schwarz) reporting the same figures, we are confident that [-30 dBc / 10 dB] is the current best wideband predistortion performance and as such, it is set to be our performance baseline.

[3] high power broadcast demands greater amplifier efficiency

For clarification, the lone [-43 dBc / 15 dB] WCDMA performance reported in the first row of Table 13.2 *logically trends* with this newly derived [-30 dBc / 10 dB] performance baseline. That is, the WCDMA case is a less difficult linearization problem with its lower ACD shoulder height and therefore it should, and indeed does, result in greater ACD shoulder attenuation. This *logical trending* builds further confidence in our derived baseline.

This baseline, to be used in the next chapter's performance testing, represents the minimum level of performance our technique must achieve if it is to be considered state-of-the-art.

Chapter 14

Performance Of The Proposed Predistortion Technique

In this chapter, performance of the proposed predistortion technique is tested on the laboratory transmitter testbed. This testbed was initially discussed in Chapter 4 and hence familiarity is assumed in the following. Performance of the *Initial Setting* optimization phase is tested first, with results compared against the previous chapter's performance baseline. Performance testing of the *On-Air Adaption* phase then follows, with full-cycle disturbance behavior being investigated. In both cases, internal optimizer parameter settings are in accordance with Section 11.4. Finally, predistortion Crest Factor growth is discussed, highlighting the need for high resolution reconstruction DACs.

14.1 *Initial Setting* Performance Testing

As discussed in Chapter 10, the *Initial Setting* optimization phase estimates the optimal predistortion filter parameters $\boldsymbol{h}_o$ at the very start of the transmitter's operational life. Its guiding optimization schedule, developed in Sections 10.4 and 11.5, is repeated in Table 14.1 for reference. In order to test the *Initial Setting* phase, the following steps are performed:

1. Initialize the predistortion filter kernel to $\boldsymbol{h} = \boldsymbol{0}$
2. Adjust the power amplifier's input power to realize a -30 dBc ACD shoulder height, consistent with the previous chapter's performance baseline
3. Measure the power amplifier output spectrum prior to predistortion
4. Implement the *Initial Setting* optimization schedule of Table 14.1
5. Measure the power amplifier output spectrum after *Initial Setting*

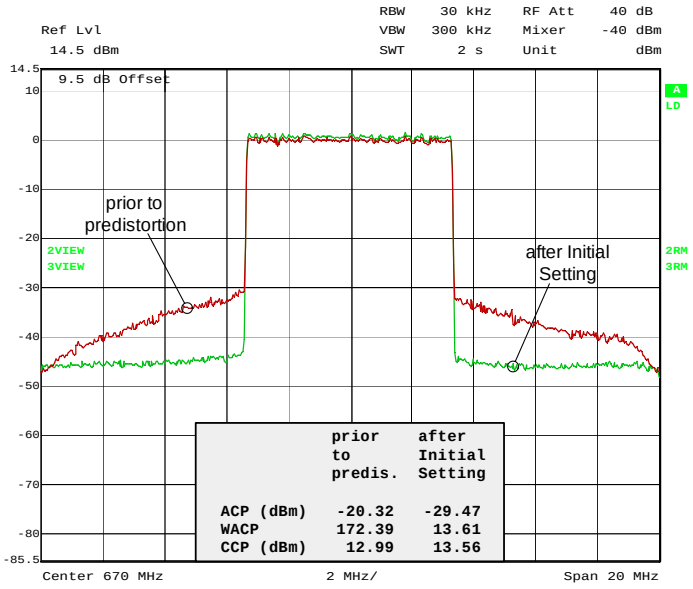
Optimization	h_s	Optimizer	WACP Weighting & Taper
1a.	$\boldsymbol{p}_{3R}$	Genetic	Quadratic, $W_E = 10^5$
1b.	$\boldsymbol{p}_{3I}$		
2a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		Quadratic, $W_E = 10^4$
2b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		
3a.	$\boldsymbol{p}_{3R}$	Nelder-Mead Simplex	Quadratic, $W_E = 10^3$
3b.	$\boldsymbol{p}_{3I}$		
4a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		Quadratic, $W_E = 10^2$
4b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		

Table 14.1: *Initial Setting* optimization schedule

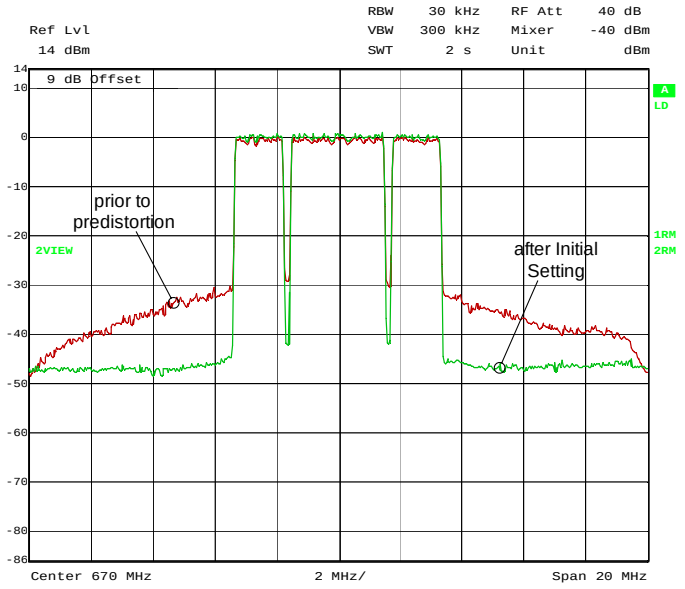
Figure 14.1 presents the output power spectra from steps 3 and 5 for each target modulation. ACP, WACP and Co-Channel Power (CCP) measurements are added to each subfigure for reference. As discussed in Chapter 3, the carrier frequency for each target modulation is set in accordance with the ACMA RF spectrum plan [13, 14] and licensed provider allocation within Australia [7, 15].

Figure 14.2 also presents the output power spectra from steps 3 and 5, but this time with slots of inband spectral power temporarily removed to uncover Co-Channel Distortion (CCD). For DVB-T and DAB, two sets of OFDM carriers are nulled during modulation whilst for WCDMA, dual notch filtering is applied directly to the modulated signal. In all cases, slot widths are kept to a minimum in an attempt to reduce signal degradation and hence maintain experimental comparison with Figure 14.1. Despite this measure being taken, slight changes in ACD are still witnessed across figures; specifically slight reduction *prior to predistortion* due to reduced amplifier drive and slight growth *after Initial Setting* due to optimality offset.

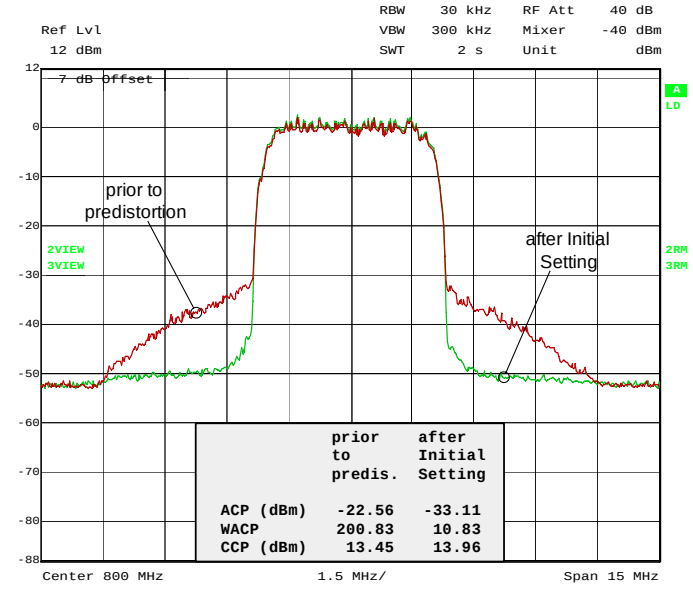
In the previous chapter, predistortion performance was quantified in the frequency-domain by the [ACD Shoulder Height, ACD Shoulder Attenuation] pair. Referring to Figure 14.1, the proposed technique is seen to achieve a performance of [-30 dBc, 13 dB] for both DVB-T and WCDMA, and [-30 dBc, 18 dB] for DAB. At first glance, these Figure 14.1 results suggest that our technique under performs for WCDMA, since its [Height, Attenuation] performance does not exceed DVB-T's and residual CCD shoulders remain after *Initial Setting*. This is not the case however for the following reasons:



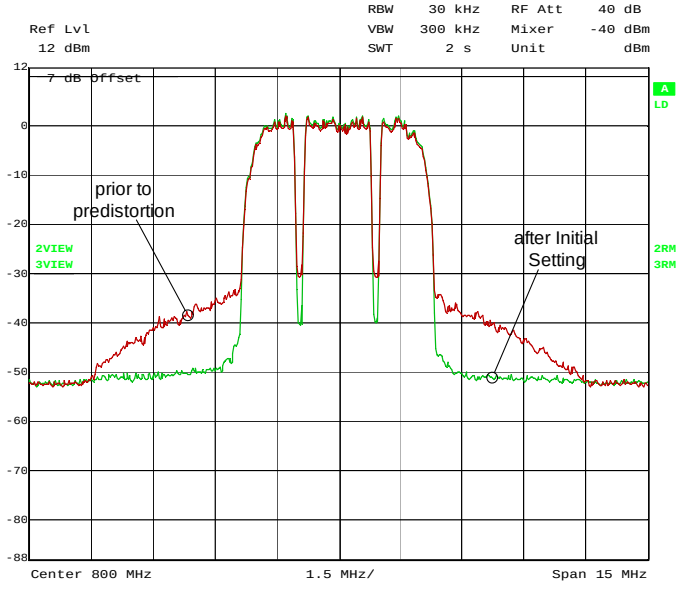
(a) DVB-T



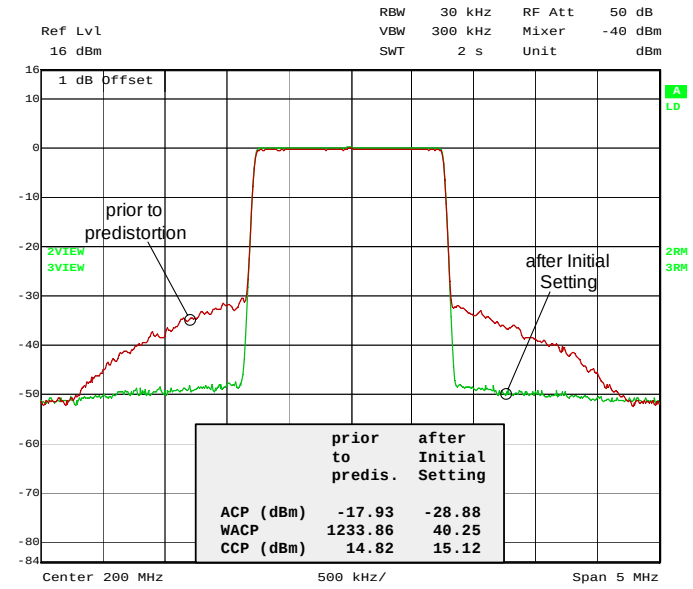
(a) DVB-T



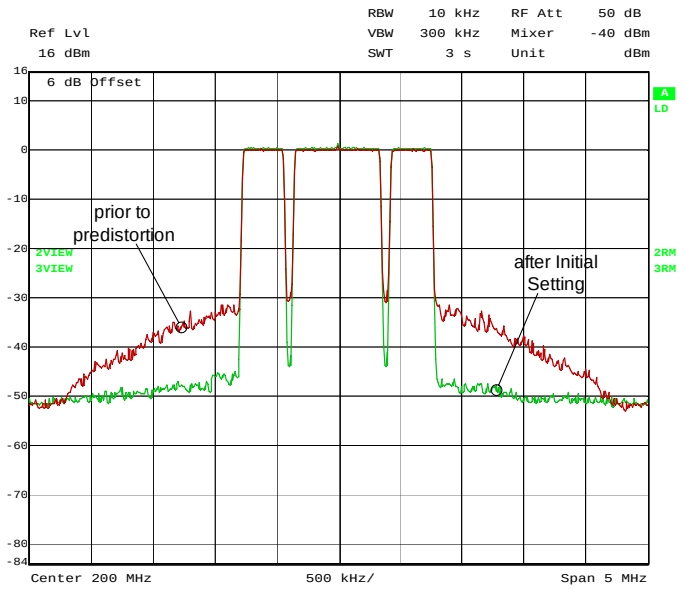
(b) WCDMA



(b) WCDMA



(c) DAB



(c) DAB

Figure 14.1: Power spectra at the testbed amplifier output prior to predistortion and after *Initial Setting* for DVB-T (a), WCDMA (b) and DAB (c).

Figure 14.2: The case of Figure 14.1 repeated, this time with slots of inband spectral power temporarily removed to uncover CCD.

1. As initially discussed in Section 4.6, our WCDMA signal comprises 512 user channels, double the standard 256. This increases the WCDMA signal's Crest Factor to that of DVB-T ($\approx$ 11.25 dB), creating a comparable predistortion problem and resulting in identical [-30 dBc, 13 dB] performance. It is only when the WCDMA signal possesses the standard number of user channels that performance can be expected to exceed that of DVB-T.

2. Despite our DVB-T and WCDMA test signals possessing comparable Crest Factors, DVB-T still possesses greater bandwidth and hence experiences greater Crest Factor growth during predistortion. In order to accommodate this greater growth, the DVB-T signal must be scaled more heavily prior to entering the reconstruction DACs. Heavier scaling, coupled with finite DAC precision, subsequently leads to a higher spectral noise floor; in our case -46 dBc for DVB-T compared to -52 dBc for WCDMA. The existence of this high noise floor is also supported by Figures 13.2, 13.3 and 13.4 (previous chapter), representing the performance of current in-service systems.

   Referring to DVB-T Subfigure 14.1a, this higher noise floor does not allow predistortion to run to completion; prematurely halting distortion reduction in the adjacent channel extremities before CCD shoulders can form, as in WCDMA Subfigure 14.1b. Whilst the noise floor is just low enough for full CCD reduction, it is not low enough for full ACD reduction and hence a flat spectral distortion trace appears.

   Put another way, if an imaginary -46 dBc noise floor was projected onto WCDMA Subfigure 14.1b, the residual CCD shoulders would be lost under the noise floor and performance would appear visually identical to that of DVB-T.[1]

   In effect, if the testbed possessed higher resolution DACs, the noise floor of Subfigure 14.1a would be lower, allowing complete predistortion, and hence the *after Initial Setting* trace would be identical to that of WCDMA; exhibiting residual CCD shoulders.

3. The existence of residual CCD shoulders in WCDMA Subfigure 14.1b is not a case of detrimental WACP *under-weighting* or *over-weighting* as initially discussed in Section 9.2. If it were a case of *under-weighting*, the CCD shoulders would be concave on the edges of the transmission band, just as the ACD is *prior to predistortion*. Alternatively, if it were a case of *over-weighting*, inner ACD scalloping would occur and additional shoulders would exist in the outer adjacent channel regions.

[1] DVB-T carrier nulling is more efficient than WCDMA notch filtering when it comes to removing inband spectral power and as such, the 1 dB difference in CCD observed between Subfigures 14.2a and 14.2b is a case of notch spectral leakage rather than difference in performance.

> These residual CCD shoulders actually represent the practical upper limits of predistortion linearization, initially spoken of in Section 6.1. Once the amplifier input drive level reaches C_I in the demonstration of Figure 6.1 (page 55), signal expansion is capped by full output saturation and distortion is inevitable.

So it is seen that when underlying user channels, DAC noise floor considerations and upper practical linearization limits are taken into account, WCDMA performance is fully validated.

In the context of the previous Dot-Point 2, it is also suspected that DAB performance, with no residual CCD shoulders present, is also limited by the spectral noise floor. Whilst the noise floor level may be lower here, so too is the difficulty of the predistortion problem and hence the same scenario is faced. In this case however, the extent of the limitation is unclear. Unlike the DVB-T case, we don't have a comparable WCDMA cross check available and hence it is quite possible that performance greater than [-30 dBc, 18 dB] is achievable if higher precision DACs are used.

In order to put the performance of this technique into context, we turn to the previous chapter's performance baseline of [-30 dBc, 10 dB]. Derived for DVB-T, the most difficult target modulation to predistort, this baseline represents the minimum level of performance our technique must achieve if it is to be considered state-of-the-art. As demonstrated in Subfigure 14.1a, our technique achieves a performance of [-30 dBc, 13 dB] for DVB-T. This is a 3 dB improvement over the baseline and as such, the proposed technique must be taken seriously.

It is important to reiterate that the spectral plots of Figures 14.1 and 14.2 are taken at the output of the power amplifier, and thus bandpass mask filtering is still to be applied. It is this additional selectivity which brings ACD levels to within spectral mask requirements. As an example, for DVB-T transmitter collocation, the Australian spectral mask is set at -48 dBc in the adjacent channel [41]. From Subfigure 14.1a, ACD after predistortion is between -46 dBc and -43 dBc, ignoring the limits of the spectral noise floor discussed earlier. As such, at the output of the amplifier, an additional 2–5 dB attenuation is required in order to satisfy the mask. With [261] reporting > 40 dB selectivity for its DVB-T bandpass filters, the mask is well satisfied prior to transmission.

Referring to the added ACP, WACP and CCP measurements of Figure 14.1, it is seen that reduction in ACP / WACP is accompanied by growth in CCP. This is not coincidental. As discussed intuitively in Section 6.1, the predistorter's reversal of signal compression leads to the restoration of top end transmitter gain and hence transmitted power.

Another observation of Figure 14.1 worthy of mention is the slight difference in lower and upper ACD shoulder heights prior to predistortion. As discussed through-

out previous chapters, this spectral asymmetry is an artifact of transmitter memory. For DVB-T, possessing greatest bandwidth, this asymmetry extends into the outer adjacent channel, with scalloping witnessed at 7 MHz above the carrier frequency.

With the presentation of linearization performance now complete, attention is turned to convergence rate performance. Table 14.2 presents the convergence rate properties of the *Initial Setting* optimization schedule for DVB-T / WCDMA / DAB. As discussed in Sections A.4 and A.5, the local *Nelder-Mead Simplex* and global *Genetic* algorithms are both robust, gradient-free strategies and hence do not require objective measurement averaging. For both algorithms and all target modulations, the spectrum analyzer sweep rate is set to two seconds.

Expectedly, the global *Genetic* algorithm is much slower to converge than the local *Nelder-Mead Simplex* algorithm, with convergence rates being measured in hours compared to minutes (last column). This makes the first half of the schedule (search phase) much slower than the last half (refinement phase). Convergence is also seen to be slowest for DVB-T and quickest for DAB. This is a direct consequence of the relative number of variables to be optimized (column 4). Total time to converge is obtained by adding the convergence rates of each optimization in the last column. For DVB-T / WCDMA / DAB, this turns out to be 25.1 / 21.8 / 9.3 hours.

It is immediately obvious from these convergence rates that the *Initial Setting* optimization phase must be left to run over night and / or in the background during the transmitter commissioning phase. This will not pose a problem however, provided technicians proactively schedule other on-site work packages to accommodate.

It is acknowledged that existing techniques have quicker convergence rates. For example the *Direct Learning* technique converges in the order of several hours [194]. However this quicker convergence rate is at the expense of linearization performance. Specifically, predistortion filter parameter estimation is a trade off between the two conflicting criteria of *convergence rate* and *linearization performance*.

Compared to existing techniques, the proposed technique occupies the opposite end of this trade off spectrum. That is, it accepts a slower convergence rate specifically to maximize linearization performance. Such performance has been demonstrated earlier in this section.

As discussed in the literature review, this trade off stems from the different parameter estimation models being employed and the validity of their assumptions relating to objective convexity. Existing techniques use a *local* linear regression model whereas the proposed technique uses a *global* generic single objective mathematical optimization model. The former cannot guarantee true global convergence whereas the latter can.

Optimization	$\boldsymbol{h_s}$	**Optimizer**	**Variables**	**Iterations**	**Objective Measurements**	**Convergence Rate**
1a.	$\boldsymbol{p}_{3R}$	Genetic	8 / 7 / 3	105 / 97 / 37	5300 / 4900 / 1900	2.9 / 2.7 / 1 hrs
1b.	$\boldsymbol{p}_{3I}$		8 / 7 / 3	As Above	As Above	As Above
2a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		24 / 21 / 9	325 / 276 / 124	16300 / 13850 / 6250	9 / 7.7 / 3.5 hrs
2b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		24 / 21 / 9	As Above	As Above	As Above
3a.	$\boldsymbol{p}_{3R}$	Nelder-Mead Simplex	8 / 7 / 3	86 / 71 / 34	227 / 177 / 75	7.5 / 6 / 2.5 mins
3b.	$\boldsymbol{p}_{3I}$		8 / 7 / 3	As Above	As Above	As Above
4a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$		24 / 21 / 9	250 / 215 / 91	951 / 762 / 244	32 / 25 / 8 mins
4b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		24 / 21 / 9	As Above	As Above	As Above
						$\Downarrow$
					Total Time To Converge:	**25.1 / 21.8 / 9.3 hrs**

Table 14.2: Convergence rate properties of the *Initial Setting* optimization schedule for DVB-T / WCDMA / DAB

14.2 *On-Air Adaption* Performance Testing

As discussed in Chapter 10, a transmitter's nonlinear transfer characteristic will drift slowly during its operational life. This is a result of component aging, temperature fluctuations and power supply voltage variations. The *On-Air Adaption* phase adapts the predistortion filter kernel during this drift, in order to maintain optimal linearization.

In practice, the *On-Air Adaption* phase is applied directly after the *Initial Setting* phase and left to run for the lifetime of the transmitter. In this sense, kernel adaption is a continuous ongoing process which constantly tracks changes in the transfer characteristic. Its guiding optimization schedule, developed in Sections 10.5 and 11.5, is repeated in Table 14.3 for reference.
Since component aging is a *long-term* process, typically occurring over a duration of years, it has minimal effect on the *short-term* operation of the *On-Air Adaption* phase. From the perspective of a continuously running optimizer, which is the *On-Air Adaption* phase, it is the *short-term* temperature fluctuations and power supply voltage variations, typically occurring over a duration of hours or days, which are the dominant causes of drift and which necessitate kernel adaption.

In the laboratory testing of the *On-Air Adaption* phase, these temperature and supply voltage fluctuations are induced to create a forced (as opposed to natural) drift. Temperature is reduced from a nominal 45℃ to 35℃ (22% change) whilst supply voltage is reduced from a nominal 28V to 25V (10.7% change) [205]. Due to the controlled environmental and electrical operating conditions of modern day transmitters [61, 194], such large changes would never be encountered in practice. They are chosen here however to test the *On-Air Adaption* phase under the most extreme drift conditions. Temperature and supply voltage are intentionally reduced in this testing in order to reduce FET conductance and compress FET dynamic range respectively, thereby severely degrading the overall nonlinearity.

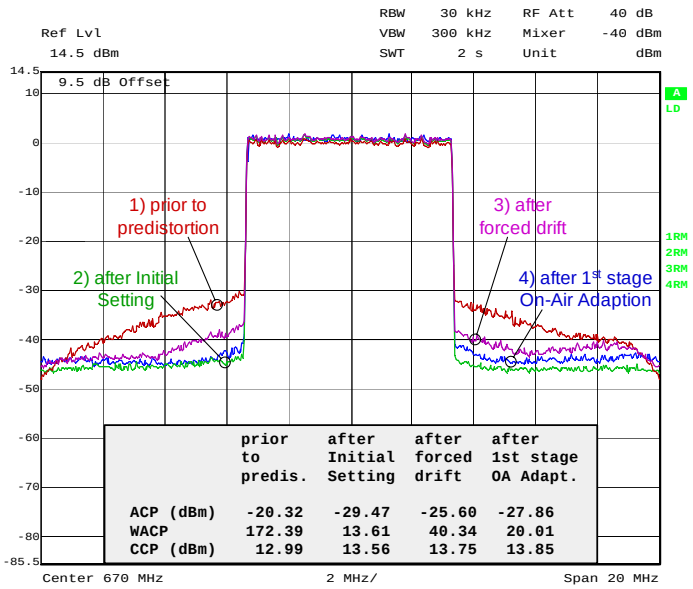
Optimization	$\boldsymbol{h}_s$	**Optimizer**	**WACP Weighting**
1a.	$\boldsymbol{p}_{3R}$		
1b.	$\boldsymbol{p}_{3I}$	Nelder-Mead	Quadratic, $W_E = 10^2$
2a.	$\left[\, \boldsymbol{p}_{5R} \mid \boldsymbol{p}_{7R} \mid \boldsymbol{p}_{9R} \,\right]$	Simplex	
2b.	$\left[\, \boldsymbol{p}_{5I} \mid \boldsymbol{p}_{7I} \mid \boldsymbol{p}_{9I} \,\right]$		
	Repeat indefinitely for constant tracking		

Table 14.3: *On-Air Adaption* optimization schedule

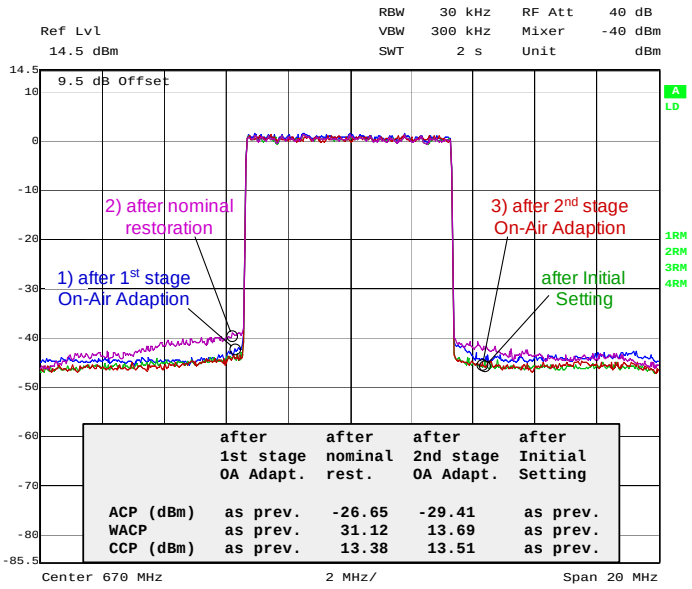
In the first stage of testing (following completion of the *Initial Setting* phase), a forced drift is applied and the *On-Air Adaption* phase is allowed to react. In the second stage of testing (continuing directly on from the first), nominal conditions are restored and the *On-Air Adaption* phase is once again allowed to react. With this two stage testing methodology, the performance of the *On-Air Adaption* phase can be observed in terms of a *full-cycle disturbance* rather than a *one-way drift*. Figures 14.3 and 14.4 present results for these first and second stages of testing respectively. Both figures are presented side-by-side on the same page to facilitate easy comparison.

In Figure 14.3 (first stage testing), output power spectra prior to predistortion, after *Initial Setting*, after forced drift and after *On-Air Adaption* are presented for each target modulation. ACP, WACP and CCP measurements are added to each subfigure for reference. The forced drift renders the *Initial Setting* predistorter kernel non-optimal, resulting in a significant increase in distortion from *Initial Setting* levels. To restore kernel optimality under the forced drift conditions, the *On-Air Adaption* phase is applied. At first glance, it appears that the *On-Air Adaption* phase under performs since its trace does not align with that of the *Initial Setting* phase. This is not the case however. From an optimization perspective, the forced drift causes the global minimum of the WACP optimization objective to not only translate in the vector space but also increase in magnitude. This increase in magnitude is a direct consequence of the amplifier's greater inherent nonlinearity as discussed previously in terms of FET conductance and dynamic range. It follows that the *On-Air Adaption* phase, under these extreme drift conditions, cannot theoretically reduce distortion back to an *Initial Setting* level. The best it can do is reduce distortion to a level corresponding to the new inferior global minimum and this is what we are observing with the two traces not matching. Difference in magnitude between the WACP global minimum before and after the forced drift is frequency and bandwidth dependent and hence *On-Air Adaption* results will vary across target modulation.

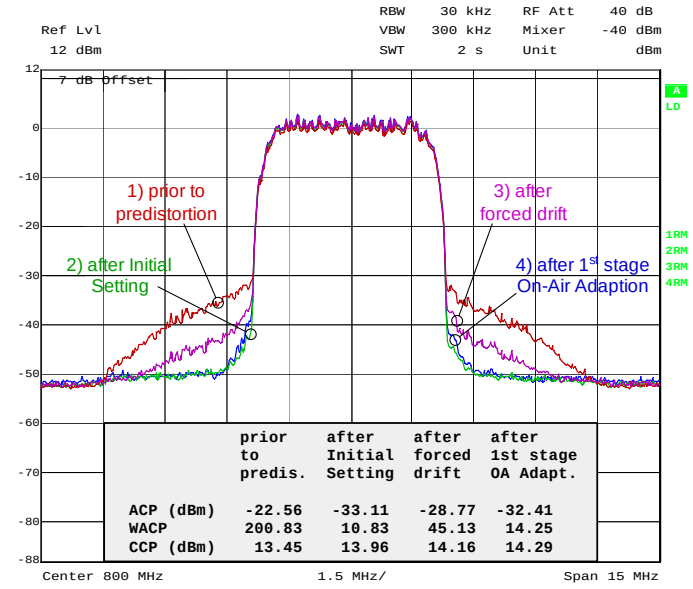
In Figure 14.4 (second stage testing), output power spectra after first stage *On-Air Adaption* (carried over from Figure 14.3), after restoration of nominal conditions and after second stage *On-Air Adaption* are presented for each target modulation. Once again, power measurements are added to each subfigure for reference. Restoration of nominal conditions renders the first stage *On-Air Adaption* predistorter kernel non-optimal, resulting in a significant increase in distortion levels. To restore kernel optimality under the nominal conditions, the second stage *On-Air Adaption* phase is applied. This brings distortion levels back to the original *Initial Setting* levels.



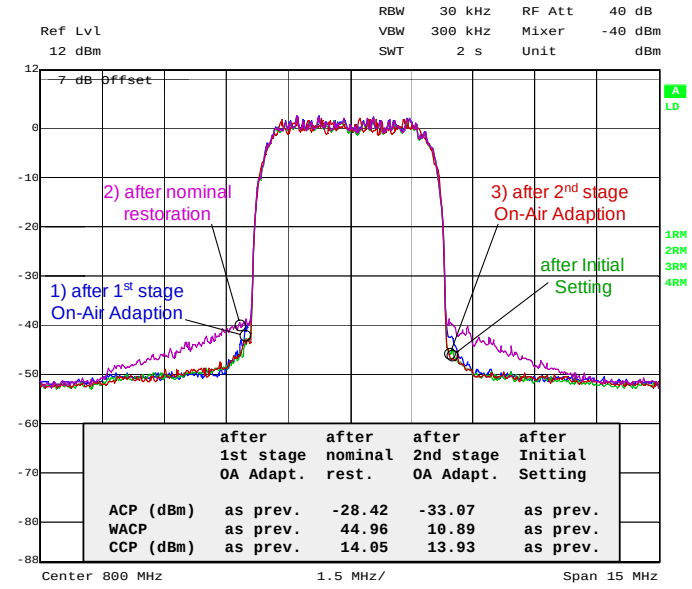
(a) DVB-T



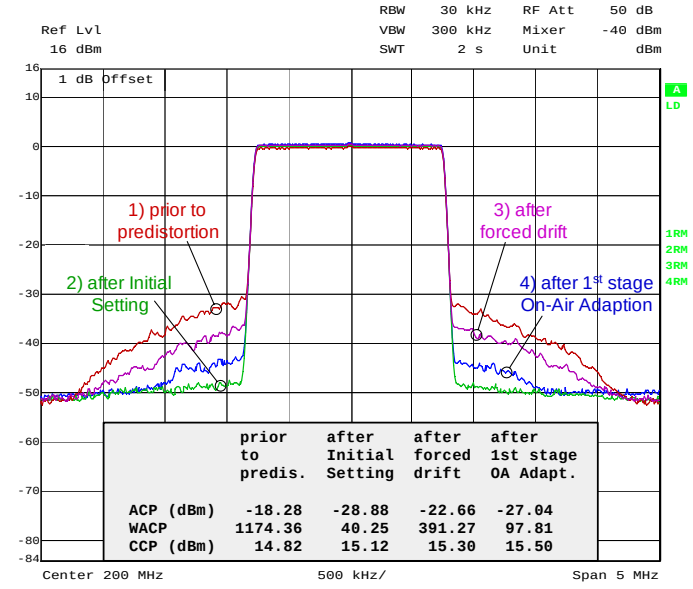
(a) DVB-T



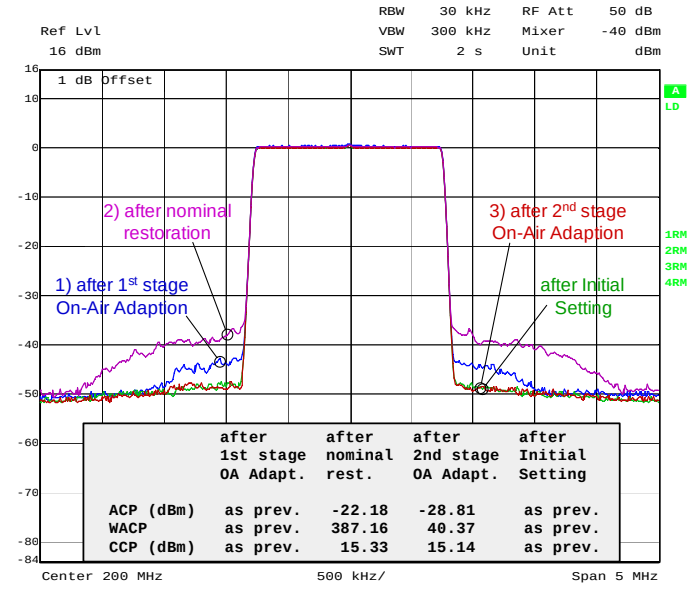
(b) WCDMA



(b) WCDMA



(c) DAB



(c) DAB

Figure 14.3: First stage of *On-Air Adaption* testing for DVB-T (a), WCDMA (b) and DAB (c). Output power spectral traces individually labeled.

Figure 14.4: Second stage of *On-Air Adaption* testing for DVB-T (a), WCDMA (b) and DAB (c). Output power spectral traces individually labeled.

The fact that distortion levels rise following a change in operating condition (first stage forced drift or second stage nominal restoration) but are subsequently reduced by application of the *On-Air Adaption* phase demonstrates the optimal tracking ability of the *On-Air Adaption* phase. The fact that the *On-Air Adaption* phase is capable of returning the predistorter kernel to its *Initial Setting* state following the full-cycle disturbance also demonstrates reliable and predictable behavior.

With the *On-Air Adaption* phase performed by the local *Nelder-Mead Simplex* algorithm, convergence rates for both first and second stage testing above are in the order of minutes for all target modulations. This rate is more than adequate, considering transmitter drift is continuous in practice with a much larger hourly / daily time frame.

Before leaving this section, Tables 14.4–14.6 present predistortion filter kernel coefficients for DVB-T, WCDMA and DAB signal modulation, following the *Initial Setting* and first stage *On-Air Adaption* phases. Three trends are immediately obvious from these tables:

1. Real components are generally an order of magnitude larger than imaginary components. This suggests that imaginary components perform a less dominant role in the dynamic memory compensation process. As discussed in Chapter 10, this offset magnitude is the reason why real components are optimized before imaginary components in both the *Initial Setting* and *On-Air Adaption* phases.

2. Coefficient magnitude reduces by several orders of magnitude per odd nonlinear order. As discussed in Chapters 10 and 11, this poor scaling instigates the development of the optimization scheduling concept and forces the *Nelder-Mead Simplex* and *Genetic* algorithms to employ coefficient pre-scaling during simplex initialization and gene mutation respectively.

3. Coefficient magnitude reduces globally when moving from DVB-T to WCDMA to DAB. This is due to a decrease in modulation bandwidth and hence total predistortion power needing to be generated for linearization.

It is also interesting to observe that, despite amplifier memory effects decaying with time, coefficient magnitudes generally hold steady with memory delay i. Considering the direct memory estimation technique of Section 12.2 and the positive predistortion performances already presented in this chapter, this phenomenon is not attributed to memory length under-estimation. Rather, what we are observing is the optimizer actively compensating for the non-dominant coefficients lost during pruning; specifically increasing and potentially phase offsetting those back end kernel coefficients which would normally decay with memory.

DVB-T KERNEL COEFFICIENTS

Kernel Order	Memory Delay i	Coefficient After Initial Setting		Coefficient After 1st Stage On-Air Adaption		Change In Coefficient	
	$M = 14, R = 2$	Real	Imag	Real	Imag	Real	Imag
	0	7.311×10^{-7}	1.645×10^{-8}	1.041×10^{-6}	2.409×10^{-7}	-3.099×10^{-7}	-2.240×10^{-7}
	1	-1.019×10^{-7}	1.739×10^{-8}	-3.000×10^{-7}	3.766×10^{-8}	1.981×10^{-7}	-2.027×10^{-8}
	2	-3.088×10^{-7}	-4.111×10^{-7}	-4.372×10^{-7}	-5.569×10^{-7}	1.284×10^{-7}	1.458×10^{-7}
3	3	-5.538×10^{-7}	1.944×10^{-8}	5.812×10^{-7}	2.713×10^{-8}	-5.538×10^{-7}	-7.690×10^{-9}
	4	5.716×10^{-7}	3.613×10^{-8}	7.607×10^{-7}	1.069×10^{-7}	-1.891×10^{-7}	-7.077×10^{-8}
	5	-2.509×10^{-7}	2.530×10^{-8}	-4.295×10^{-7}	-3.926×10^{-8}	1.786×10^{-7}	6.456×10^{-8}
	6	-2.901×10^{-7}	-9.742×10^{-8}	-3.554×10^{-7}	-1.023×10^{-8}	6.530×10^{-8}	-8.719×10^{-8}
	7	2.456×10^{-8}	3.697×10^{-8}	3.270×10^{-7}	7.074×10^{-8}	-3.024×10^{-7}	-3.377×10^{-8}
	0	-5.481×10^{-13}	-1.000×10^{-13}	1.513×10^{-13}	-5.865×10^{-14}	-6.994×10^{-13}	-4.135×10^{-14}
	1	-3.861×10^{-13}	-2.402×10^{-14}	-3.741×10^{-13}	-6.835×10^{-14}	-1.200×10^{-14}	4.433×10^{-14}
	2	1.150×10^{-13}	-2.515×10^{-14}	9.218×10^{-14}	-4.759×10^{-14}	2.282×10^{-14}	2.244×10^{-14}
5	3	-2.941×10^{-13}	2.070×10^{-15}	3.102×10^{-13}	4.271×10^{-15}	-6.043×10^{-13}	-2.201×10^{-15}
	4	-8.944×10^{-14}	-5.892×10^{-13}	-3.091×10^{-13}	-2.351×10^{-14}	2.197×10^{-13}	-5.657×10^{-13}
	5	2.964×10^{-13}	-7.423×10^{-14}	-5.351×10^{-14}	4.122×10^{-14}	3.499×10^{-13}	-1.155×10^{-13}
	6	1.943×10^{-13}	5.576×10^{-14}	8.593×10^{-14}	-9.292×10^{-15}	1.084×10^{-13}	6.505×10^{-14}
	7	-3.518×10^{-14}	-4.955×10^{-14}	7.336×10^{-13}	-3.258×10^{-14}	-7.688×10^{-13}	-1.697×10^{-14}
	0	4.528×10^{-19}	3.986×10^{-20}	5.667×10^{-19}	8.842×10^{-20}	-1.139×10^{-19}	-4.856×10^{-20}
	1	-1.181×10^{-19}	2.599×10^{-19}	-2.630×10^{-19}	-3.558×10^{-20}	1.449×10^{-19}	2.955×10^{-19}
	2	3.392×10^{-19}	-3.370×10^{-20}	3.587×10^{-19}	-5.138×10^{-20}	-1.950×10^{-20}	1.768×10^{-20}
7	3	1.108×10^{-19}	2.973×10^{-19}	1.583×10^{-19}	2.896×10^{-19}	-4.750×10^{-20}	7.700×10^{-21}
	4	-4.831×10^{-19}	5.032×10^{-20}	-4.965×10^{-19}	9.633×10^{-20}	1.340×10^{-20}	-4.601×10^{-20}
	5	-8.751×10^{-19}	2.458×10^{-20}	-3.276×10^{-19}	9.935×10^{-20}	-5.475×10^{-19}	-7.477×10^{-20}
	6	4.435×10^{-20}	-1.032×10^{-19}	6.833×10^{-19}	2.425×10^{-20}	-6.390×10^{-19}	-1.275×10^{-19}
	7	3.366×10^{-19}	7.425×10^{-20}	-9.523×10^{-20}	-4.476×10^{-20}	4.318×10^{-19}	1.191×10^{-19}
	0	4.617×10^{-24}	2.270×10^{-24}	4.947×10^{-24}	2.260×10^{-24}	-3.300×10^{-25}	1.000×10^{-26}
	1	-5.556×10^{-24}	-2.854×10^{-25}	-5.378×10^{-24}	-2.983×10^{-25}	-1.780×10^{-25}	1.290×10^{-26}
	2	6.532×10^{-24}	-3.229×10^{-25}	3.232×10^{-24}	-3.525×10^{-25}	3.300×10^{-24}	2.960×10^{-26}
9	3	2.028×10^{-24}	1.010×10^{-25}	2.086×10^{-24}	1.304×10^{-25}	-5.800×10^{-26}	-2.940×10^{-26}
	4	-7.416×10^{-25}	-1.816×10^{-25}	9.858×10^{-25}	-3.805×10^{-24}	-1.727×10^{-24}	3.623×10^{-24}
	5	-5.725×10^{-24}	3.665×10^{-24}	-5.438×10^{-24}	-2.488×10^{-25}	-2.870×10^{-25}	3.914×10^{-24}
	6	3.887×10^{-24}	8.256×10^{-25}	3.375×10^{-24}	-2.574×10^{-25}	5.120×10^{-25}	1.083×10^{-24}
	7	9.736×10^{-25}	-3.943×10^{-25}	1.454×10^{-24}	7.466×10^{-25}	-4.804×10^{-25}	-1.141×10^{-24}

Table 14.4: Predistortion filter kernel coefficients following *Initial Setting* and first stage *On-Air Adaption* for DVB-T.

WCDMA KERNEL COEFFICIENTS

Kernel Order	Memory Delay i	Coefficient After Initial Setting		Coefficient After 1st Stage On-Air Adaption		Change In Coefficient	
	$M = 20, R = 3$	Real	Imag	Real	Imag	Real	Imag
	0	7.845×10^{-5}	1.675×10^{-6}	1.021×10^{-4}	2.133×10^{-5}	-2.365×10^{-5}	-1.966×10^{-5}
	1	1.291×10^{-5}	-1.020×10^{-5}	-7.173×10^{-5}	-6.892×10^{-6}	8.464×10^{-5}	-3.308×10^{-6}
	2	-2.963×10^{-5}	-5.636×10^{-6}	-4.878×10^{-5}	-1.135×10^{-5}	1.915×10^{-5}	5.714×10^{-6}
3	3	2.406×10^{-5}	-2.320×10^{-6}	4.808×10^{-5}	5.301×10^{-6}	-2.402×10^{-5}	-7.621×10^{-6}
	4	-5.390×10^{-7}	9.610×10^{-6}	2.349×10^{-5}	6.339×10^{-7}	-2.403×10^{-5}	8.976×10^{-6}
	5	7.266×10^{-5}	-5.997×10^{-6}	-4.694×10^{-6}	-2.424×10^{-6}	7.735×10^{-5}	-3.573×10^{-6}
	6	3.475×10^{-6}	7.462×10^{-7}	5.358×10^{-5}	-3.337×10^{-6}	-5.011×10^{-5}	4.083×10^{-6}
	0	-8.780×10^{-9}	-2.809×10^{-10}	-3.576×10^{-9}	-9.327×10^{-11}	-5.204×10^{-9}	-1.876×10^{-10}
	1	9.950×10^{-9}	-5.558×10^{-10}	4.724×10^{-9}	-3.148×10^{-10}	5.226×10^{-9}	-2.410×10^{-10}
	2	1.166×10^{-9}	-1.230×10^{-9}	-4.836×10^{-10}	-3.696×10^{-9}	1.650×10^{-9}	2.466×10^{-9}
5	3	-6.614×10^{-9}	2.639×10^{-10}	-7.551×10^{-9}	7.799×10^{-11}	9.370×10^{-10}	1.859×10^{-10}
	4	-1.629×10^{-9}	5.182×10^{-11}	2.217×10^{-9}	-1.878×10^{-10}	-3.846×10^{-9}	2.396×10^{-10}
	5	-7.978×10^{-10}	-1.347×10^{-10}	-4.321×10^{-10}	-4.181×10^{-10}	-3.657×10^{-10}	2.834×10^{-10}
	6	3.546×10^{-10}	-4.335×10^{-11}	7.336×10^{-10}	-2.743×10^{-10}	-3.790×10^{-10}	2.310×10^{-10}
	0	-1.780×10^{-13}	9.245×10^{-14}	2.064×10^{-12}	5.794×10^{-13}	-2.242×10^{-12}	-4.870×10^{-13}
	1	1.581×10^{-12}	2.532×10^{-13}	5.063×10^{-12}	5.434×10^{-13}	-3.482×10^{-12}	-2.902×10^{-13}
	2	8.150×10^{-13}	4.743×10^{-13}	2.957×10^{-12}	1.074×10^{-12}	-2.142×10^{-12}	-5.997×10^{-13}
7	3	-2.071×10^{-12}	1.520×10^{-13}	1.300×10^{-12}	9.635×10^{-13}	-3.371×10^{-12}	-8.115×10^{-13}
	4	2.315×10^{-12}	5.207×10^{-13}	3.152×10^{-12}	1.515×10^{-13}	-8.370×10^{-13}	3.692×10^{-13}
	5	6.182×10^{-13}	6.323×10^{-13}	1.399×10^{-12}	1.934×10^{-11}	-7.808×10^{-13}	-1.871×10^{-11}
	6	4.476×10^{-13}	2.894×10^{-14}	3.836×10^{-12}	8.449×10^{-13}	-3.388×10^{-12}	-8.160×10^{-13}
	0	9.482×10^{-16}	1.482×10^{-17}	3.471×10^{-16}	5.285×10^{-16}	6.011×10^{-16}	-5.137×10^{-16}
	1	-1.602×10^{-16}	-1.124×10^{-16}	5.170×10^{-16}	4.201×10^{-17}	-6.772×10^{-16}	-1.544×10^{-16}
	2	-7.352×10^{-16}	-3.624×10^{-17}	-5.935×10^{-16}	-3.827×10^{-17}	-1.417×10^{-16}	2.030×10^{-18}
9	3	-2.483×10^{-16}	-6.708×10^{-17}	-1.066×10^{-15}	-2.575×10^{-16}	8.177×10^{-16}	1.904×10^{-16}
	4	-7.842×10^{-17}	-8.686×10^{-17}	-9.434×10^{-16}	-2.629×10^{-16}	8.650×10^{-16}	1.760×10^{-16}
	5	7.294×10^{-16}	-2.039×10^{-16}	-2.107×10^{-16}	1.543×10^{-17}	9.401×10^{-16}	-2.193×10^{-16}
	6	3.774×10^{-16}	1.527×10^{-17}	9.251×10^{-16}	-2.644×10^{-16}	-5.477×10^{-16}	-2.491×10^{-16}

Table 14.5: Predistortion filter kernel coefficients following *Initial Setting* and first stage *On-Air Adaption* for WCDMA.

DAB KERNEL COEFFICIENTS

Kernel Order	**Memory Delay i**	**Coefficient After Initial Setting**		**Coefficient After 1st Stage On-Air Adaption**		**Change In Coefficient**	
	$M = 14, R = 5$	Real	Imag	Real	Imag	Real	Imag
	0	1.780×10^{-5}	-2.703×10^{-6}	1.230×10^{-5}	1.490×10^{-5}	2.050×10^{-5}	-1.760×10^{-5}
3	1	-6.580×10^{-5}	1.051×10^{-6}	3.581×10^{-6}	-6.448×10^{-7}	-6.938×10^{-5}	1.696×10^{-6}
	2	-6.569×10^{-6}	-6.534×10^{-7}	-7.822×10^{-5}	1.362×10^{-6}	7.165×10^{-5}	-2.015×10^{-6}
	0	5.185×10^{-9}	3.232×10^{-10}	3.739×10^{-9}	5.284×10^{-10}	1.446×10^{-9}	-2.052×10^{-10}
5	1	6.506×10^{-9}	1.257×10^{-10}	5.207×10^{-9}	5.309×10^{-10}	1.299×10^{-9}	-4.052×10^{-10}
	2	3.142×10^{-9}	-4.799×10^{-11}	1.810×10^{-9}	-1.489×10^{-10}	1.332×10^{-9}	1.009×10^{-10}
	0	-1.537×10^{-12}	8.552×10^{-13}	-4.093×10^{-12}	1.129×10^{-12}	2.556×10^{-12}	-2.738×10^{-13}
7	1	1.586×10^{-13}	-3.739×10^{-13}	2.157×10^{-12}	-8.125×10^{-13}	-1.999×10^{-12}	4.386×10^{-13}
	2	-9.163×10^{-12}	-8.692×10^{-13}	1.146×10^{-12}	-9.573×10^{-13}	-1.031×10^{-11}	8.810×10^{-14}
	0	2.230×10^{-16}	8.809×10^{-17}	9.939×10^{-16}	9.422×10^{-17}	-7.709×10^{-16}	-6.130×10^{-18}
9	1	9.032×10^{-17}	-7.966×10^{-17}	2.524×10^{-16}	4.586×10^{-17}	-1.621×10^{-16}	-1.255×10^{-16}
	2	1.577×10^{-16}	9.282×10^{-17}	2.496×10^{-17}	1.426×10^{-16}	1.327×10^{-16}	-4.978×10^{-17}

Table 14.6: Predistortion filter kernel coefficients following *Initial Setting* and first stage *On-Air Adaption* for DAB.

14.3 Crest Factor Growth

As discussed in Section 6.1, a digital predistortion system fundamentally implements a signal expansion characteristic in order to compensate for the power amplifier's impending compression. This expansion characteristic logically leads to signal Crest Factor growth. Taking into account memory, this growth increases with bandwidth. Figure 14.5 demonstrates this growth for DVB-T, in terms of the Complementary Cumulative Distribution Function (CCDF). A signal's CCDF represents the probability (y-axis) of the signal's instantaneous power exceeding its mean power by a specified value (x-axis). As such, the CCDF x-intercept represents Crest Factor.

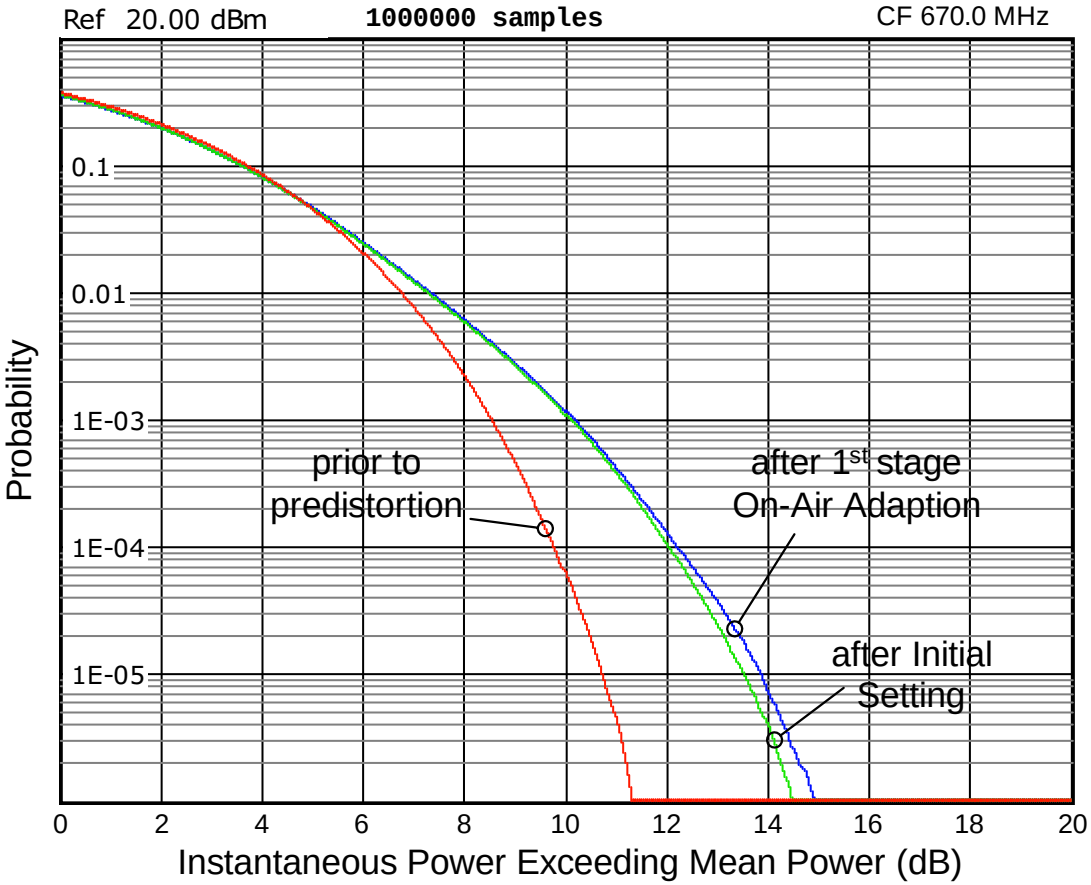


Figure 14.5: CCDF of the predistortion filter output signal prior to predistortion, after *Initial Setting* and after first stage *On-Air Adaption* for DVB-T modulation. Derived using the envelope power approach and 1 000 000 signal samples [18].

It is seen here that the predistortion filter output signal prior to predistortion is a pure DVB-T signal with 11.25 dB Crest Factor. After *Initial Setting* and first stage *On-Air Adaption* however, the predistortion filter expands signal peaks and increases Crest Factor by 3.25 dB and 3.75 dB respectively, to counteract the downstream

compression of the power amplifier. It is noted that signal expansion is greater in the first stage *On-Air Adaption* case since power amplifier compression is more severe following the forced drift.

Crest Factor growth requires headroom preallocation along the exciter stage path and hence places greater expectations on the dynamic range of exciter components. Finite precision reconstruction DACs are generally the weakest link in this respect, as is the case in the laboratory transmitter testbed.

In terms of headroom preallocation, testbed DAC inputs are scaled such that peak excursion is approximately 30%, 36% and 39% of DAC full scale for DVB-T, WCDMA and DAB respectively. This staggered scaling accomodates the different Crest Factor growths associated with bandwidth. After *Initial Setting* and first stage *On-Air Adaption*, peak excursion for all target modulations grows to approximately 75% and 80% of DAC full scale respectively, and hence destructive clipping is avoided.

In terms of dynamic range, testbed DACs possess a reasonable 14-bit resolution. With the above scaling and Crest Factor growth however, only a small portion of these bits are utilized on the average and hence spectral noise floors are driven up to -46 dBc, -52 dBc and -51 dBc for DVB-T, WCDMA and DAB respectively. As discussed earlier in Section 14.1, these noise floors are found to interfere with DVB-T and DAB bottom end predistortion performance and hence testbed DACs turn out to be several bits short of ideal.

It follows that special consideration must be given to DAC resolution in predistortion systems. What may be adequate for normal transmission will unlikely be adequate for predistortion transmission, considering the high Crest Factors involved. In this sense, it is always recommended to use the highest precision DACs available.

This chapter has demonstrated the performance of the proposed predistortion technique on real hardware, specifically the laboratory transmitter testbed. Initial Setting results show a 3 dB improvement in performance over what is considered current state-of-the-art. Full-cycle On-Air Adaption results also show optimal tracking ability and predictable behavior in beyond worst case disturbance conditions. The need for high resolution reconstruction DACs is also reinforced in the context of dynamic range and hence performance potential.

Chapter 15

Summary & Conclusion

Digital predistortion is a modern linearization technique allowing transmitters to be operated more efficiently and hence cost effectively. In accordance with our Statement of Research in Section 3.1, this thesis has specifically demonstrated adaptive digital predistortion for wideband high Crest Factor applications based on the concept of *Spectral Power Feedback Learning (SPFL)*. Prior to this work, *SPFL* had only been proposed for narrowband, linearly modulated systems.

Unlike the current generation *Direct* and *Indirect Learning* strategies, which rely on *time-domain signal feedback*, the *SPFL* strategy operates with *frequency-domain information feedback*. This makes the strategy more suited to current and future wideband applications since temporal feedback delay and gain compensation is avoided. This is consistent with demonstrated state-of-the-art performance results obtained from real hardware testing of DVB-T, WCDMA and DAB signal modulations.

Research Contributions

With the predistortion filter's characterizing parameters interpreted as a set of variables to be optimized and a measure of output spectral distortion interpreted as a linearizing optimization objective, the *SPFL* strategy employs generic mathematical optimization to estimate predistortion filter parameters. In the context of this optimization framework, this research's wideband evolution of the *SPFL* strategy has produced the following contributions to the digital predistortion community:

1. the definition of a new spectral distortion optimization objective, specifically the *Weighted Adjacent Channel Power (WACP)*. In addition to conveying complete adjacent channel behavior, this objective is able to discriminate between spectral distortion components and hence control the location of spectral distortion reduction. This makes linearization more robust in the presence of residual memory effects.

2. the definition of a new predistortion filter architecture to accommodate amplifier memory effects. Based on hybrid triple-stage pruning of the classic Baseband Volterra Series, this architecture possesses a dynamically matched kernel whose size is not only linear with respect to memory, but also independent of hardware sampling rate implementation. Ultimately, this kernel size ensures a practically manageable optimization vector space and hence improved convergence reliability.

3. the demonstration of a new experimental procedure for estimating predistortion filter memory. This procedure is based on sweep-probing the transmitter with memory specific distortion created by the predistortion filter, and looking for changes in the signature of the output ACP spectrum. Compared to traditional approaches, this procedure is considered more direct and accurate since it estimates memory directly with the predistortion filter in place and hence replicates what the predistortion filter and transmitter would experience in practice during optimization.

4. the derivation of *Initial Setting* and *On-Air Adaption* optimization schedules based on the concept of *influential subsets*. This concept ensures optimization is always performed over the *minimally sized variable vector which guarantees complete observability* and hence optimization convergence reliability is always maximal.

5. the development of the *Distortion Array*; a graphical organizing tool for keeping track of nonlinear distortion components generated by the predistorter-amplifier cascade. Using this tool, all facets of the predistortion process can be described both visually and intuitively in the time-domain; this includes the fundamental concepts of *parasitic* growth, nonlinear order interaction and theoretical limitation. Without the *Distortion Array*, predistortion concepts are easily lost in mathematical rigor.

The legitimacy of these research contributions is confirmed with two full-length, peer reviewed journal articles being published in the IEEE Transactions on Broadcasting.

The Way Forward

Considering its demonstrated state-of-the-art performance, we believe the way forward for this technique is technology commercialization. With the assistance of Uniquest (James Cook University's commercialization business), we have filed an international patent application and passed the international search report phase; highlighting our claims of novelty. We are now seeking commercialization partners and subsequent funding to progress the patent from provisional to full status.

The commercialization partners we intend targeting are digital broadcast and mobile basestation transmitter manufacturers; specifically Rohde & Schwarz, NEC, Harris, Ericsson and Nokia Siemens. In approaching these global companies, our discussions will be focused on the following key points:

- future growth in modulation bandwidth will further expose the feedback weaknesses of current generation *Direct* and *Indirect Learning* strategies, leading to reduced linearization performance. The proposed technique on the other hand, with its novel predistortion filter *architecture* and *parameter estimation strategy*, is far more robust and well suited to future wideband applications.

- the proposed technique's performance is demonstrated on a working hardware testbed, eliminating any uncertainty in the validity of results. Furthermore, this demonstrated performance is shown to be state-of-the-art when baselined against current in-service wideband predistortion systems.

Based on the solid research leading to this point, we approach the upcoming commercialization effort with vigor and optimism. We truly believe this technique has the potential to advance digital predistortion application and represent the next generation of predistortion system.

Bibliography

[1] *Technical Specification Group Radio Access Network; Base Station (BS) Conformance Testing (FDD)*, 3rd Generation Partnership Project (3GPP) Technical Specification TS 25.141 V9.0.0 (2009-05). [Online]. Available: www.3gpp.org

[2] *Technical Specification Group Radio Access Network; Base Station (BS) Radio Transmission and Reception (FDD)*, 3rd Generation Partnership Project (3GPP) Technical Specification TS 25.104 V8.5.0 (2008-12). [Online]. Available: www.3gpp.org

[3] G. Acciari, F. Giannini, E. Limiti, and M. Rossi, "Baseband predistorter using direct spline computation," in *Proc. IEE Circuits Devices Systems*, vol. 152, no. 3, Jun. 2005, pp. 259–265.

[4] C. S. Adjiman and C. A. Floudas, "Rigorous convex underestimators for general twice-differentiable problems," *Journal of Global Optimization*, no. 9, pp. 23–40, 1996.

[5] "Understanding CDMA measurements for base stations and their components," Application Note 1311, Agilent Technologies, pp. 25–27, 2000. [Online]. Available: www.agilent.com

[6] "Testing and troubleshooting digital RF communications transmitter designs," Application Note 1313, Agilent Technologies, January 2002. [Online]. Available: www.agilent.com

[7] A. Alderson, P. Gallagher, E. Gregory, and B. Lazzaro, "Telstra's Next G^{TM}: pretty in PIM," *Stay Connected: The Radio Frequency Systems Bulletin*, pp. 7–9, Second Quarter 2007. [Online]. Available: www.rfsworld.com

[8] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, ser. Information and System Sciences. New Jersey: Prentice Hall, 1979.

[9] S. Andreoli, H. G. McClure, P. Banelli, and S. Cacopardi, "Digital linearizer for RF amplifiers," *IEEE Transactions on Broadcasting*, vol. 43, no. 1, pp. 12–19, Mar. 1997.

[10] I. P. Androulakis, C. D. Maranas, and C. A. Floudas, "αBB: A global optimization method for general constrained nonconvex problems," *Journal of Global Optimization*, vol. 4, no. 7, pp. 337–363, 1995.

[11] H. Anton, *Elementary Linear Algebra*, 8th ed. New Jersey: John Wiley & Sons, 2000.

[12] *Microwave Office / Analog Office User Guide*, Version 7.5, Applied Wave Research (AWR), El Segundo, June 2007. [Online]. Available: www.appwave.com

[13] "Australian radiofrequency spectrum allocations chart," Australian Communications and Media Authority (ACMA), Canberra, January 2009. [Online]. Available: www.acma.gov.au

[14] "Australian radiofrequency spectrum plan," Australian Communications and Media Authority (ACMA), Canberra, January 2009. [Online]. Available: www.acma.gov.au

[15] "Radio and television broadcasting stations: Internet edition," Australian Communications and Media Authority (ACMA), Canberra, February 2011. [Online]. Available: www.acma.gov.au

[16] T. Bäck, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. New York: Oxford University Press, January 1996.

[17] T. Bäck, D. B. Fogel, and Z. Michalewicz, Eds., *Handbook of Evolutionary Computation*, ser. Computational Intelligence Library. New York: Oxford University Press, April 1997.

[18] C. Balz, "CCDF determination - a comparison of two measurement methods," *News From Rohde & Schwarz*, no. 172, pp. 52–53, 2001. [Online]. Available: www.rohde-schwarz.com

[19] M. Banerjee, "Level accuracy and electronic level settings of SMIQ," Application Note, Rohde & Schwarz, Munich, May 2001. [Online]. Available: www.rohde-schwarz.com

[20] N. A. Barricelli, "Numerical testing of evolution theories: Part I - theoretical introduction and basic tests," *Acta Biotheoretica*, vol. 16, no. 1/2, pp. 69–98, March 1962.

[21] ——, "Numerical testing of evolution theories: Part II - preliminary tests of performance (symbiogenesis and terrestrial life)," *Acta Biotheoretica*, vol. 16, no. 3/4, pp. 99–126, September 1963.

[22] G. Baudoin and P. Jardin, "Adaptive polynomial pre-distortion for linearization of power amplifiers in wireless communications and WLAN," in *International Conference on Trends in Communications, EUROCON'01*, vol. 1, Jul., pp. 157–160.

[23] M. G. Bellanger, *Adaptive Digital Filters*, 2nd ed., ser. Signal Processing and Communications. New York: Marcel Dekker, 2001.

[24] N. Benvenuto, F. Piazza, and A. Uncini, "A neural network approach to data predistortion with memory in digital radio systems," in *IEEE International Conference on Communications*, vol. 1, Geneva, May 1993, pp. 232–236.

[25] E. Biglieri, S. Barberis, and M. Catena, "Analysis and compensation of nonlinearities in digital transmission systems," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 1, pp. 42–51, Jan. 1988.

[26] H. Black, "Translating system," U.S. Patent 1-686-792, Oct. 9, 1928.

[27] ——, U.S. Patent 2-102-671, Dec., 1937.

[28] W. Bösch and G. Gatti, "Measurement and simulation of memory effects in predistortion linearizers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 37, no. 12, pp. 1885–1890, Dec. 1989.

[29] S. Boumaiza and F. M. Ghannouchi, "Thermal memory effects modeling and compensation in RF power amplifiers and predistortion linearizers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, no. 12, pp. 2427–2433, Dec. 2003.

[30] A. Carini, E. Mumolo, and G. L. Sicuranza, "V-Vector algebra and its application to Volterra-adaptive filtering," *IEEE Transactions on Circuits and Systems—Part II: Analog and Digital Signal Processing*, vol. 46, no. 5, pp. 585–598, May 1999.

[31] J. K. Cavers, "Amplifier linearization using a digital predistorter with fast adaptation and low memory requirements," *IEEE Transactions on Vehicular Technology*, vol. 39, no. 4, pp. 374–382, Nov. 1990.

[32] ——, "A linearizing predistorter with fast adaptation," in *IEEE 40th Vehicular Technology Conference*, May 1990, pp. 41–47.

[33] ——, "The effect of quadrature modulator and demodulator errors on adaptive digital predistorters for amplifier linearization," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 2, pp. 456–466, May 1997.

[34] J. Cha, I. Kim, S. Hong, B. Kim, J. S. Lee, and H. S. Kim, "Memory effect minimization and wide instantaneous bandwidth operation of a base station power amplifier," *Microwave Journal*, vol. 50, no. 1, pp. 66–76, Jan. 2007.

[35] S. Chang and E. J. Powers, "A simplified predistorter for compensation of nonlinear distortion in OFDM systems," in *IEEE Global Telecommunications Conference, GLOBECOM'01*, vol. 5, 2001, pp. 3080–3084.

[36] S. Chang, E. J. Powers, and J. Chung, "A compensation scheme for nonlinear distortion in OFDM systems," in *IEEE Global Telecommunications Conference, GLOBECOM'00*, vol. 2, 2000, pp. 736–740.

[37] C.-H. Cheng and E. J. Powers, "Optimal Volterra kernel estimation algorithms for a nonlinear communication system for PSK and QAM inputs," *IEEE Transactions on Signal Processing*, vol. 49, no. 1, pp. 147–163, Jan. 2001.

[38] C. K. Chui and G. Chen, *Kalman Filtering With Real-Time Applications*, 3rd ed., ser. Information Sciences. Berlin: Springer, 1999.

[39] C. J. Clark, G. Chrisikos, M. S. Muha, A. A. Moulthrop, and C. P. Silva, "Time-domain envelope measurement technique with application to wideband power amplifier modeling," *IEEE Transactions on Microwave Theory and Techniques*, vol. 46, no. 12, pp. 2531–2540, Dec. 1998.

[40] G. W. Collins, *Fundamentals of Digital Television Transmission.* New York: John Wiley & Sons, 2001.

[41] *Digital Television - Terrestrial Broadcasting, Part 1: Characteristics of Digital Terrestrial Television Transmissions*, Council of Standards Australia, Committee CT-002 Broadcasting and Related Services Std. AS 4599.1-2007, May 2007.

[42] D. Cox, "Linear amplification using nonlinear components," *IEEE Transactions on Communications*, vol. COM-22, pp. 1942–1945, Dec. 1974.

[43] P. Cramer and Y. Rolaine, "Broadband measurement and identification of a Wiener-Hammerstein model for an RF amplifier," in *60th ARFTG Conference Digest*, Dec. 2002, pp. 49–57.

[44] C. Crespo-Cadenas, Universidad de Sevilla, Spain, Dec. 2008, private email communication.

[45] C. Crespo-Cadenas, J. Reina-Tosina, and M. J. Madero-Ayora, "Volterra behavioral model for wideband RF amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 55, no. 3, pp. 449–457, Mar. 2007.

[46] S. C. Cripps, *RF Power Amplifiers for Wireless Communications*, ser. Microwave. Massachusetts: Artech House, 1999.

[47] J. Czech, "A linearized L-Band 200 Watt TWT amplifier for multicarrier operation," in *16th European Microwave Conference*, Sep. 1986, pp. 810–815.

[48] W. Dai, P. Roblin, and M. Frei, "Distributed and multiple time-constant electro-thermal modeling and its impact on ACPR in RF predistortion," in *62nd ARFTG Microwave Measurements Conference*, Dec. 2003, pp. 89–98.

[49] A. N. D'Andrea and V. Lottici, "RF power amplifier linearization through amplitude and phase predistortion," *IEEE Transactions on Communications*, vol. 44, no. 11, Nov. 1996.

[50] A. N. D'Andrea, V. Lottici, and R. Reggiannini, "A digital approach to efficient RF power amplifier linearization," in *IEEE Global Telecommunications Conference, GLOBECOM'97*, vol. 1, Nov. 1997, pp. 77–81.

[51] W. C. Davidon, "Variable metric method for minimization," *SIAM Journal on Optimization*, vol. 1, no. 1, pp. 1–17, Feb. 1991.

[52] N. B. de Carvalho and J. C. Pedro, "Two-tone IMD asymmetry in microwave power amplifiers," in *Microwave Symposium Digest, IEEE MTT-S International*, Jun. 2000, pp. 445–448.

[53] ——, "A comprehensive explanation of distortion sideband asymmetries," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 9, pp. 2090–2101, Sep. 2002.

[54] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, ser. Classics In Applied Mathematics. Philadelphia: SIAM, 1996.

[55] L. Ding, Z. Ma, D. R. Morgan, M. Zierdt, and J. Pastalan, "A least-squares/Newton method for digital predistortion of wideband signals," *IEEE Transactions on Communications*, vol. 54, no. 5, pp. 833–840, May 2006.

[56] L. Ding, R. Raich, and G. T. Zhou, "A Hammerstein predistortion linearization design based on the indirect learning architecture," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'02*, vol. 3, May 2002, pp. 2689–2692.

[57] L. Ding, G. T. Zhou, D. R. Morgan, Z. Ma, J. S. Kenney, J. Kim, and C. R. Giardina, "Memory polynomial predistorter based on the indirect learning architecture," in *IEEE Global Telecommunications Conference, GLOBECOM'02*, vol. 1, Nov. 2002, pp. 967–971.

[58] ——, "A robust digital baseband predistorter constructed using memory polynomials," *IEEE Transactions on Communications*, vol. 52, no. 1, pp. 159–165, Jan. 2004.

[59] M. Djamai, S. Bachir, and C. Duvanaud, "Behavioral modeling and digital predistortion of RF power amplifiers," in *International Workshop on Integrated Nonlinear Microwave and Millimeter-Wave Circuits*, Jan. 2006, pp. 160–163.

[60] N. Dye and H. Granberg, *Radio Frequency Transistors: Principles and Practical Applications*, 2nd ed. Boston: Newnes, 2001.

[61] *Radio Base Station (RBS) 2206 Reference Manual*, EN/LZT 720 008 R1A, Ericsson Radio Systems, Stockholm, Jun. 2001.

[62] C. Eun and E. J. Powers, "A predistorter design for a memory-less nonlinearity preceded by a dynamic linear system," in *Global Telecommunications Conference, GLOBECOM'95*, Nov. 1995, pp. 152–156.

[63] ——, "A new Volterra predistorter based on the indirect learning architecture," *IEEE Transactions on Signal Processing*, vol. 45, no. 1, pp. 223–227, Jan. 1997.

[64] *Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television*, European Telecommunications Standards Institute (ETSI) Std. EN 300 744 V1.4.1 (2001-01). [Online]. Available: www.etsi.org

[65] *Radio Broadcasting Systems; Digital Audio Broadcasting (DAB) To Mobile, Portable and Fixed Receivers*, European Telecommunications Standards Institute (ETSI) Std. EN 300 401 V1.4.1 (2006-01). [Online]. Available: www.etsi.org

[66] *Digital Video Broadcasting (DVB); Measurement Guidelines For DVB Systems*, European Telecommunications Standards Institute (ETSI) Technical Report ETR 290, May 1997. [Online]. Available: www.etsi.org

[67] M. Faulkner and M. Johansson, "Adaptive linearization using predistortion - experimental results," *IEEE Transactions on Vehicular Technology*, vol. 43, no. 2, pp. 323–332, May 1994.

[68] M. Faulkner and T. Mattsson, "Spectral sensitivity of power amplifiers to quadrature modulator misalignment," *IEEE Transactions on Vehicular Technology*, vol. 41, no. 4, pp. 516–525, Nov. 1992.

[69] M. Faulkner, T. Mattsson, and W. Yates, "Adaptive linearisation using predistortion," in *IEEE 40th Vehicular Technology Conference*, May 1990, pp. 35–40.

[70] K. Fazel and S. Kaiser, *Multi-Carrier and Spread Spectrum Systems*. West Sussex: John Wiley & Sons, 2003.

[71] G. Feng, L. M. Li, and S. Q. Wu, "A modified adaptive compensation scheme for nonlinear bandlimited satellite channels," in *Global Telecommunications Conference, GLOBECOM'91*, vol. 3, Dec. 1991, pp. 1551–1555.

[72] W. Fischer, *Digital Video and Audio Broadcasting Technology: A Practical Engineering Guide*, 2nd ed., ser. Signals and Communication Technology. Berlin: Springer-Verlag, 2008.

[73] ——, "DVB-T/H transmitter measurements for acceptance, operation and monitoring," Application Note 04.209-7BM101-OE, Rohde & Schwarz, Munich, 2009. [Online]. Available: www.rohde-schwarz.com

[74] R. Fletcher, *Practical Methods of Optimization*, 2nd ed. West Sussex: John Wiley & Sons, 1987.

[75] C. A. Floudas, *Deterministic Global Optimization: Theory, Methods and Applications*, ser. Nonconvex Optimization And Its Applications. Dordrecht: Kluwer Academic, 2000, vol. 37.

[76] A. S. Fraser, "Simulation of genetic systems by automatic digital computers," *Australian Journal of Biological Science*, vol. 10, pp. 484–491, 1957.

[77] K. Gentile, "The care and feeding of digital, pulse-shaping filters," *RF Design*, pp. 50–61, April 2002. [Online]. Available: www.rfdesign.com

[78] M. Ghaderi, S. Kumar, and D. E. Dodds, "Adaptive predistortion lineariser using polynomial functions," *IEE Proceedings - Communications*, vol. 141, no. 2, pp. 49–55, Apr. 1994.

[79] ——, "Fast adaptive polynomial I and Q predistorter with global optimization," *IEE Proceedings - Communications*, vol. 143, no. 2, pp. 78–86, Apr. 1996.

[80] F. M. Ghannouchi and O. Hammi, "Behavioral modeling and predistortion," *IEEE Microwave Magazine*, pp. 52–64, Dec. 2009.

[81] P. L. Gilabert, G. Montoro, and E. Bertran, "On the Wiener and Hammerstein models for power amplifier predistortion," in *Asia-Pacific Microwave Conference, APMC'05*, vol. 2, Dec. 2005, p. 4.

[82] P. L. Gilabert, D. D. Silveira, G. Montoro, M. E. Gadringer, and E. Bertran, "Heuristic algorithms for power amplifier behavioral modeling," *IEEE Microwave and Wireless Components Letters*, vol. 17, no. 10, pp. 715–717, Oct. 2007.

[83] P. L. Gilabert, D. D. Silveira, G. Montoro, and G. Magerl, "RF-power amplifier modeling and predistortion based on a modular approach," in *European Microwave Integrated Circuits Conference*, Sep. 2006, pp. 265–268.

[84] P. L. Gilabert, E. Bertran, G. Montoro, and J. Berenguer, "FPGA implementation of an LMS-based real-time adaptive predistorter for power amplifiers," in *Joint IEEE North-East Workshop on Circuits and Systems and TAISA Conference, NEWCAS-TAISA'09*, Jul., p. 4.

[85] P. L. Gilabert, A. Cesari, G. Montoro, E. Bertran, and J.-M. Dilhac, "Multi-lookup table FPGA implementation of an adaptive digital predistorter for linearizing RF power amplifiers with memory effects," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 2, pp. 372–384, Feb. 2008.

[86] H. Girard and K. Feher, "A new baseband linearizer for more efficient utilization of earth station amplifiers used for QPSK transmission," *IEEE Journal on Selected Areas in Communications*, vol. 1, no. 1, pp. 46–56, Jan. 1983.

[87] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning.* Boston: Addison-Wesley Longman, January 1989.

[88] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Maryland: Johns Hopkins University Press, 1996.

[89] J. Grabowski and R. C. Davis, "An experimental M-QAM modem using amplifier linearization and baseband equalization techniques," in *National Telesystems Conference, NTC'82, Conference Record (A84-15623 04-32)*, Galveston TX, Nov. 1982, paper E3.2, pp. 1–6.

[90] L. F. Gray, J. van Alstyne, and W. A. Sandrin, "Application of broadband linearizers to satellite earth stations," in *International Conference on Communications, ICC'80, Conference Record (A81-32276 14-32)*, Seattle WA, Jun. 1980, paper 33.4, pp. 1–5.

[91] A. Grebennikov, *RF and Microwave Power Amplifier Design*, ser. Electronic Engineering. New York: McGraw-Hill, 2005.

[92] W. Greblicki, "Nonparametric identification of Wiener systems," *IEEE Transactions on Information Theory*, vol. 38, no. 5, pp. 1487–1493, Sep. 1992.

[93] W. Greblicki and M. Pawlak, "Nonparametric identification of Hammerstein systems," *IEEE Transactions on Information Theory*, vol. 35, no. 2, pp. 409–418, Mar. 1989.

[94] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB*, 2nd ed. New York: Wiley-Interscience, 2001.

[95] S. Grunwald, "Measurements on MPEG2 and DVB-T signals - part 3," News From Rohde & Schwarz - Number 170, Munich, 2001. [Online]. Available: www.rohde-schwarz.com

[96] O. Hammi, S. Boumaiza, M. Jaïdane-Saïdane, and F. M. Ghannouchi, "Digital subband filtering predistorter architecture for wireless transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 5, pp. 1643–1652, May 2005.

[97] O. Hammi and F. M. Ghannouchi, "Reduced complexity Volterra models for nonlinear system identification," *EURASIP Journal on Advances in Signal Processing*, vol. 4, pp. 257–265, Jul. 2001.

[98] ——, "Twin nonlinear two-box models for power amplifiers and transmitters exhibiting memory effects with application to digital predistortion," *IEEE Microwave and Wireless Components Letters*, vol. 19, no. 8, pp. 530–532, Aug. 2009.

[99] O. Hammi, F. M. Ghannouchi, and B. Vassilakis, "A compact envelope-memory polynomial for RF transmitters modeling with application to baseband and RF-digital predistortion," *IEEE Microwave and Wireless Components Letters*, vol. 18, no. 5, pp. 359–361, May 2008.

[100] L. Hanzo and T. Keller, *OFDM and MC-CDMA: A Primer*. West Sussex: John Wiley & Sons, 2006.

[101] S. Hara and R. Prasad, *Multicarrier Techniques For 4G Mobile Communications*, ser. Universal Personal Communications. London: Artech House, 2003.

[102] "Maxiva ULX Liquid-Cooled UHF Multimedia TV Transmitter," Data Sheet, Harris Corporation, Mason OH, 2011. [Online]. Available: www.broadcast.harris.com

[103] "Platinum VLX VHF Liquid-Cooled TV/DAB Transmitter," Data Sheet, Harris Corporation, Mason OH, 2012. [Online]. Available: www.broadcast.harris.com

[104] S. Haykin, *Adaptive Filter Theory*, 4th ed., ser. Information and System Sciences, T. Kailath, Ed. New Jersey: Prentice Hall, 2002.

[105] M. Helaoui, S. Boumaiza, A. Ghazel, and F. M. Ghannouchi, "On the RF/DSP design for efficiency of OFDM transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 7, pp. 2355–2361, Jul. 2005.

[106] M. M. Hella and M. Ismail, *RF CMOS Power Amplifiers - Theory, Design and Implementation*, ser. Engineering and Computer Science. New York: Kluwer International, 2002.

[107] D. Hilborn, S. P. Stapleton, and J. K. Cavers, "An adaptive direct conversion transmitter," vol. 43, no. 2, pp. 223–233, May 1992.

[108] W. Hoeg and T. Lauterbach, Eds., *Digital Audio Broadcasting: Principles and Applications*. Chichester: John Wiley & Sons, 2001.

[109] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Ann Arbor: University of Michigan Press, 1975.

[110] W. Holz, "An IF-Predistorter for TWTA-Linearization in 16 QAM digital radio," in *14th European Microwave Conference*, Sep. 1984, pp. 543–548.

[111] R. Horst, P. M. Pardalos, and N. V. Thoai, *Introduction to Global Optimization*, 2nd ed., ser. Nonconvex Optimization and Its Applications. Dordrecht: Kluwer Academic, 2000.

[112] S. Im and E. J. Powers, "An application of a digital predistortion linearizer to CDMA HPA's," in *Global Telecommunications Conference, GLOBECOM'04*, vol. 6, Dec. 2004, pp. 3907–3910.

[113] M. Isaksson, "Radio frequency power amplifiers: Behavioral modeling, parameter-reduction and digital predistortion," Ph.D. dissertation, KTH School of Electrical Engineering, Stockholm Sweden, 2007.

[114] M. Isaksson and D. Rönnow, "Digital predistortion of radio frequency power amplifiers using Kautz-Volterra model," *Electronics Letters*, vol. 42, no. 13, Jun. 2006.

[115] ——, "A Kautz-Volterra behavioral model for RF power amplifiers," in *Microwave Symposium Digest, IEEE MTT-S International*, Jun. 2006, pp. 485–488.

[116] M. Isaksson and D. Wisell, "Extension of the Hammerstein model for power amplifier applications," in *63rd ARFTG Conference Digest*, Jun. 2004, pp. 131–137.

[117] M. Isaksson, D. Wisell, and D. Rönnow, "A comparative analysis of behavioral models for RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, no. 1, pp. 348–359, Jan. 2006.

[118] G. James, *Advanced Modern Engineering Mathematics*, 2nd ed. Essex: Addison-Wesley, 1999.

[119] E. G. Jeckeln, F. M. Ghannouchi, and M. A. Sawan, "An L band adaptive digital predistorter for power amplifiers using direct I-Q modem," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 2, Jun. 1998, pp. 719–722.

[120] ——, "A new adaptive predistortion technique using software-defined radio and DSP technologies suitable for base station 3G power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 9, pp. 2139–2147, Sep. 2004.

[121] V. John Mathews and G. L. Sicuranza, *Polynomial Signal Processing*, ser. Telecommunications and Signal Processing. New York: John Wiley & Sons, 2000.

[122] C. Johnson, *Radio Access Networks for UMTS: Principles and Practice*. West Sussex: John Wiley & Sons, 2008.

[123] N. M. Josuttis, *The C++ Standard Library: A Tutorial and Reference*. Boston: Addison-Wesley, 1999.

[124] L. Kahn, "Single-sideband transmission by envelope elimination and restoration," in *Proc. IRE*, vol. 40, Jul. 1952, pp. 803–806.

[125] H. W. Kang, Y. S. Cho, and D. H. Youn, "An efficient adaptive predistorter for nonlinear high power amplifier in satellite communication," in *IEEE International Symposium on Circuits and Systems*, vol. 4, Jun. 1997, pp. 2288–2291.

[126] ——, "On compensating nonlinear distortions of an OFDM system using an efficient adaptive predistorter," *IEEE Transactions on Communications*, vol. 47, no. 4, pp. 522–526, Apr. 1999.

[127] G. Karam and H. Sari, "Analysis of predistortion, equalization, and ISI cancellation techniques in digital radio systems with nonlinear transmit amplifiers," *IEEE Transactions on Communications*, vol. 37, no. 12, pp. 1245–1253, Dec. 1989.

[128] ——, "Data predistortion techniques using intersymbol interpolation," *IEEE Transactions on Communications*, vol. 38, no. 10, pp. 1716–1723, Oct. 1990.

[129] ——, "A data predistortion technique with memory for QAM radio systems," *IEEE Transactions on Communications*, vol. 39, no. 2, pp. 336–344, Feb. 1991.

[130] M. R. Karim and M. Sarraf, *W-CDMA and CDMA2000 for 3G Mobile Networks*. New York: McGraw-Hill, 2002.

[131] A. Katz, "Linearization: Reducing distortion in power amplifiers," *IEEE Microwave Magazine*, pp. 37–49, Dec. 2001.

[132] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, ser. Frontiers in Applied Mathematics. Philadelphia: SIAM, 1995.

[133] ——, "Detection and remediation of stagnation in the Nelder-Mead algorithm using a sufficient decrease condition," *SIAM Journal on Optimization*, no. 10, pp. 43–55, 1999.

[134] ——, *Iterative Methods for Optimization*, ser. Frontiers In Applied Mathematics. Philadelphia: SIAM, 1999, no. 18.

[135] P. B. Kenington, *High Linearity RF Amplifier Design*, ser. Microwave. Massachusetts: Artech House, 2000.

[136] ——, "Linearized transmitters: An enabling technology for software defined radio," *IEEE Communications Magazine*, vol. 40, pp. 156–162, Feb. 2002.

[137] A. M. Khilla, J. Schaufler, D. Leucht, and A. Born, "A compact wideband Ku-Band linearizer for satellite transmit amplifiers," in *19th European Microwave Conference*, Sep. 1989, pp. 706 –712.

[138] J. Kim and K. Konstantinou, "Digital predistortion of wideband signals based on power amplifier model with memory," *Electronics Letters*, vol. 37, no. 23, pp. 1417–1418, Nov. 2001.

[139] J. Kim, Y. Y. Woo, J. Moon, and B. Kim, "A new wideband adaptive digital predistortion technique employing feedback linearization," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 2, pp. 385–392, Feb. 2008.

[140] M.-C. Kim, Y. Shin, and S. Im, "Compensation of nonlinear distortion using a predistorter based on the fixed point approach in OFDM systems," in *48th IEEE Vehicular Technology Conference*, vol. 3, May 1998, pp. 2145–2149.

[141] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983.

[142] J. Korhonen, *Introduction to 3G Mobile Communications*, 2nd ed., ser. Mobile Communications. Massachusetts: Artech House, 2003.

[143] E. Kreyszig, *Advanced Engineering Mathematics*, 8th ed. New York: John Wiley & Sons, 1999.

[144] H. Ku, "Behavioral modeling of nonlinear RF power amplifiers for digital wireless communication systems with implications for predistortion linearization systems," Ph.D. dissertation, Georgia Institute of Technology, Oct. 2003.

[145] H. Ku, M. D. McKinley, and J. S. Kenney, "Extraction of accurate behavioral models for power amplifiers with memory effects using two-tone measurements," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 1, 2002, pp. 139 –142.

[146] ——, "Quantifying memory effects in RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 12, pp. 2843–2849, Dec. 2002.

[147] H. Ku and J. Stevenson Kenney, "Behavioral modeling of nonlinear RF power amplifiers considering memory effects," *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, no. 12, pp. 2495–2504, Dec. 2003.

[148] J. W. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the Nelder-Mead simplex algorithm in low dimensions," *SIAM Journal on Optimization*, no. 9, pp. 112–147, 1998.

[149] R. E. Larson, R. P. Hostetler, and B. H. Edwards, *Calculus*, 6th ed. Boston: Houghton Mifflin, 1998.

[150] G. Lazzarin, S. Pupolin, and A. Sarti, "Nonlinearity compensation in digital radio systems," *IEEE Transactions on Communications*, vol. 42, no. 2, pp. 988–999, Feb. 1994.

[151] N. Le Gallou, E. Ngoya, H. Burêt, D. Barataud, and J. M. Nébus, "An improved behavioral modeling technique for high power amplifiers with memory," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 2, 2001, pp. 983–986.

[152] K. C. Lee and P. Gardner, "Comparison of different adaptation algorithms for adaptive digital predistortion based on EDGE standard," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 2, 2001, pp. 1353 –1356.

[153] ——, "Adaptive neuro-fuzzy inference system (ANFIS) digital predistorter for RF power amplifier linearization," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 1, pp. 43–51, Jan. 2006.

[154] P. D. Leedy and J. E. Ormrod, *Practical Research: Planning and Design*, 8th ed. New Jersey: Pearson, 2005.

[155] H. Li, D. Wang, Z. Chen, and N. Liu, "Behavioural modelling of power amplifiers with memory effects based on subband decomposition," *Electronics Letters*, vol. 43, no. 5, p. 2, Mar. 2007.

[156] J. Li and J. Ilow, "A least-squares Volterra predistorter for compensation of non-linear effects with memory in OFDM transmitters," in *3rd Annual Communication Networks and Services Research Conference*, May 2005, pp. 197–202.

[157] ——, "Adaptive Volterra predistorters for compensation of non-linear effects with memory in OFDM systems," in *4th Annual Communication Networks and Services Research Conference*, May 2006, p. 4.

[158] Y. Li and P. H. Yang, "Data predistortion with adaptive fuzzy systems," in *IEEE International Conference on Systems, Man and Cybernetics, SMC'99*, vol. 6, 1999, pp. 168–172.

[159] Y. H. Lim, Y. S. Cho, I. W. Cha, and D. H. Youn, "Adaptive nonlinear prefilter for compensation of distortion in nonlinear systems," *IEEE Transactions on Signal Processing*, vol. 46, no. 6, pp. 1726–1730, Jun. 1998.

[160] E. G. Lima, T. R. Cunha, H. M. Teixeira, M. Pirola, and J. C. Pedro, "Baseband derived Volterra series for power amplifier modeling," in *Microwave Symposium Digest, IEEE MTT-S International*, Jun. 2009, pp. 1361–1364.

[161] L. Litwin and M. Pugel, "The principles of OFDM," *RF Design*, pp. 30–48, January 2001. [Online]. Available: www.rfdesign.com

[162] T. Liu, S. Boumaiza, and F. M. Ghannouchi, "Dynamic behavioral modeling of 3G power amplifiers using real-valued time-delay neural networks," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 3, pp. 1025–1033, Mar. 2004.

[163] ——, "Deembedding static nonlinearities and accurately identifying and modeling memory effects in wide-band RF transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 11, pp. 3578–3587, Nov. 2005.

[164] ——, "Augmented Hammerstein predistorter for linearization of broad-band wireless transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, no. 4, pp. 1340–1349, Apr. 2006.

[165] T. Liu, S. Boumaiza, A. B. Sesay, and F. M. Ghannouchi, "Dynamic nonlinear behavior characterization for wideband RF transmitters using augmented Hammerstein models," in *Asia-Pacific Microwave Conference, APMC'06*, Dec. 2006, pp. 967–970.

[166] ——, "Quantitative measurements of memory effects in wideband RF power amplifiers driven by modulated signals," *IEEE Microwave and Wireless Components Letters*, vol. 17, no. 1, pp. 79–81, Jan. 2007.

[167] D. G. Luenberger, *Optimization By Vector Space Method*. New York: John Wiley & Sons, 1969.

[168] A. Luzzatto and G. Shirazi, *Wireless Transceiver Design*. West Sussex: John Wiley & Sons, 2007.

[169] J. Makhoul, "Linear prediction: A tutorial review," in *Proceedings of The IEEE*, vol. 63, no. 4, Apr. 1975, pp. 561–580.

[170] G. Mandyam and J. Lai, *Third-Generation CDMA Systems For Enhanced Data Services*, ser. Communications, Networking and Multimedia. New York: Academic Press, 2002.

[171] J. A. Manner, *Spectrum Wars - The Policy and Technology Debate*, ser. Telecommunications. Massachusetts: Artech House, 2003.

[172] P. Manninen, "Effect of feedback delay error on adaptive digital predistortion," *Electronics Letters*, vol. 35, no. 14, pp. 1124–1126, Jul. 1999.

[173] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and Adaptive Signal Processing*, ser. Signal Processing. Norwood: Artech House, 2005.

[174] A. Mansell and A. Bateman, "Practical implementation issues for adaptive predistortion transmitter linearisation," *IEE Colloquium on Linear RF Amplifiers and Transmitters*, Apr. 1994.

[175] C. D. Maranas and C. A. Floudas, "Global minimum potential energy conformations of small molecules," *Journal of Global Optimization*, no. 4, pp. 135–170, 1994.

[176] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, Jun. 1963.

[177] R. Marsalek, P. Jardin, and G. Baudoin, "From post-distortion to predistortion for power amplifiers linearization," *IEEE Communications Letters*, vol. 7, no. 7, pp. 308–310, Jul. 2003.

[178] J. P. Martins, P. M. Cabral, N. B. Carvalho, and J. C. Pedro, "A metric for the quantification of memory effects in power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, no. 12, pp. 4432–4439, Dec. 2006.

[179] *Matlab Optimization Toolbox User's Guide*, Version 4.2, The Mathworks, Natick, March 2009. [Online]. Available: www.mathworks.com

[180] G. L. Matthaei, L. Young, and E. M. T. Jones, *Microwave Filters, Impedance-Matching Networks and Coupling Structures*, ser. Microwave. Massachusetts: Artech House, 1980.

[181] K. Mekechuk, W. J. Kim, S. P. Stapleton, and J. H. Kim, "Linearizing power amplifiers using digital predistortion, EDA tools and test hardware," *High Frequency Electronics*, pp. 18–27, Apr. 2004.

[182] M. Minowa, M. Onoda, E. Fukuda, and Y. Daido, "Backoff improvement of an 800-MHz GaAs FET amplifier for a QPSK transmitter using an adaptive nonlinear distortion canceller," in *IEEE 40th Vehicular Technology Conference*, May 1990, pp. 542–546.

[183] G. Montoro, P. L. Gilabert, E. Bertran, A. Cesari, and J. A. Garcia, "An LMS-based adaptive predistorter for cancelling nonlinear memory effects in RF power amplifiers," in *Asia-Pacific Microwave Conference, 2007. APMC 2007.*, Dec. 2007, p. 4.

[184] G. Montoro, P. L. Gilabert, E. Bertran, A. Cesari, and D. D. Silveira, "A new digital predictive predistorter for behavioral power amplifier linearization," *IEEE Microwave and Wireless Components Letters*, vol. 17, no. 6, pp. 448–450, Jun. 2007.

[185] J. J. Móre and D. C. Sorensen, "Computing a trust region step," *SIAM Journal on Scientific and Statistical Computing*, vol. 4, no. 3, pp. 553–572, September 1983.

[186] D. R. Morgan, Z. Ma, J. Kim, M. G. Zierdt, and J. Pastalan, "A generalized memory polynomial model for digital predistortion of RF power amplifiers," *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 3852–3860, Oct. 2006.

[187] S. D. Muruganathan and A. B. Sesay, "A QRD-RLS-Based predistortion scheme for high-power amplifier linearization," *IEEE Transactions on Circuits and Systems—Part II: Express Briefs*, vol. 53, no. 10, pp. 1108–1112, Oct. 2006.

[188] S. K. Myoung, D. Chaillot, P. Roblin, W. Dai, and S. J. Doo, "Volterra characterization and predistortion linearization of multi-carrier power amplifiers," in *64th ARFTG Microwave Measurements Conference*, Dec. 2004, pp. 65–73.

[189] Y. Nagata, "Linear amplification technique for digital mobile communications," in *IEEE 39th Vehicular Technology Conference*, vol. 1, May 1989, pp. 159–164.

[190] S. W. Nam and E. J. Powers, "On the linearization of Volterra nonlinear systems using third-order inverses in the digital frequency domain," in *IEEE International Symposium on Circuits and Systems*, May 1990, pp. 407–410.

[191] J. Namiki, "An automatically controlled predistorter for multilevel quadrature amplitude modulation," *IEEE Transactions on Communications*, vol. 31, no. 5, pp. 707–712, May 1983.

[192] M. Nannicini, P. Magni, and F. Oggionni, "Temperature controlled predistortion circuits for 64 QAM microwave power amplifiers," *Microwave Symposium Digest, IEEE MTT-S International*, pp. 99–102, Jun. 1985.

[193] *2.5kW VHF Digital TV Transmitter Instruction Manual Vol. I*, DTV-40/2R5PQ, NEC Corporation, Tokyo.

[194] *5kW UHF Digital TV Transmitter Instruction Manual Vol. I*, DTU-31/5R0PQ, NEC Corporation, Tokyo.

[195] "DTU-52 Series High-Power Digital TV Transmitter," Data Sheet, NEC Corporation, Tokyo, 2010, Cat. No. H01-10030015E 2nd Ed. [Online]. Available: www.nec.com/global/prod/nw/broadcast

[196] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 8, pp. 308–313, 1965.

[197] A. Neumaier, "Interval methods for systems of equations," in *Encyclopedia of Mathematics and its Applications*. New York: Cambridge University Press, 1990.

[198] M. K. Nezami, "Fundamentals of power amplifier linearization using digital pre-distortion," *High Frequency Electronics*, pp. 54–59, Sep. 2004. [Online]. Available: www.highfrequencyelectronics.com

[199] E. Ngoya, N. Le Gallou, J. M. Nébus, H. Burêt, and P. Reig, "Accurate RF and microwave system level modeling of wideband nonlinear circuits," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 1, 2000, pp. 79–82.

[200] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., ser. Operations Research, T. V. Mikosch, S. I. Resnick, and S. M. Robinson, Eds. New York: Springer, 2006.

[201] T. Nojima and T. Konno, "Cuber predistortion linearizer for relay equipment in 800 MHz band land mobile telephone system," *IEEE Transactions on Vehicular Technology*, vol. 34, no. 4, pp. 169–177, Nov. 1985.

[202] B. O'Brien, J. Dooley, A. Zhu, and T. J. Brazil, "Estimation of memory length for RF power amplifier behavioral models," in *36th European Microwave Conference*, Sep. 2006, pp. 680–682.

[203] M. O'Droma, E. Bertran, J. Portilla, N. Mgebrishvili, S. D. Guerrieri, G. Montoro, T. J. Brazil, and G. Magerl, "On linearisation of microwave-transmitter solid-state power amplifiers," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 15, no. 5, pp. 491–505, Sep. 2005.

[204] S. O'Leary, *Understanding Digital Terrestrial Broadcasting*, ser. Digital Audio and Video. Massachusetts: Artech House, 2000.

[205] "Solid state broadband high power RF amplifier: Model 5303038," Data Sheet, Ophir RF, Los Angeles. [Online]. Available: www.ophirrf.com

[206] H. Paaso and A. Mämmelä, "Comparison of direct learning and indirect learning predistortion architectures," in *IEEE International Symposium on Wireless Communication Systems, ISWCS'08*, Oct., pp. 309–313.

[207] I.-S. Park and E. J. Powers, "A new predistorter design technique for nonlinear digital communication channels," in *Fourth International Symposium on Signal Processing and Its Applications, ISSPA'96*, vol. 2, Aug. 1996, pp. 618–621.

[208] K. J. Parsons, P. B. Kenington, and J. P. McGeehan, "Efficient linearisation of RF power amplifiers for wideband applications," in *IEE Colloquium on Linear RF Amplifiers and Transmitters*, Apr. 1994, p. 7.

[209] J. C. Pedro and S. A. Maas, "A comparative overview of microwave and wireless power-amplifier behavioral modeling approaches," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 4, pp. 1150–1163, Apr. 2005.

[210] J. C. Pedro and N. B. Carvalho, *Intermodulation Distortion In Microwave and Wireless Circuits*. Massachusetts: Artech House, 2003.

[211] V. Petrovic, "Reduction of spurious emission from radio transmitters by means of modulation feedback," in *Proc. IEE Radio Spectrum Conservation Techniques*, Sep. 1983, pp. 44–49.

[212] C. Potter, "System analysis of a W-CDMA base-station PA employing adaptive digital predistortion," in *IEEE Radio Frequency Integrated Circuits (RFIC) Symposium*, 2002, pp. 275–278.

[213] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. New York: Cambridge University Press, 2007.

[214] S. Pupolin, A. Sarti, and H. Fu, "Performance analysis of digital radio links with nonlinear transmit amplifier and data predistorter with memory," in *IEEE International Conference on Communications, ICC'89*, Jun. 1989, pp. 292–296.

[215] S. Pupolin and L. J. Greenstein, "Digital radio performance when the transmitter spectral shaping follows the power amplifier," *IEEE Transactions on Communications*, vol. 35, no. 3, pp. 261–266, Mar. 1987.

[216] ——, "Performance analysis of digital radio links with nonlinear transmit amplifiers," *IEEE Journal on Selected Areas in Communications*, vol. SAC-5, no. 3, pp. 534–546, Apr. 1987.

[217] F. H. Raab, P. Asbeck, S. Cripps, P. B. Kenington *et al.*, "RF and microwave power amplifier and transmitter technologies - part 3," *High Frequency Electronics*, pp. 34–48, Sep. 2003. [Online]. Available: www.highfrequencyelectronics.com

[218] F. H. Raab, P. Asbeck, S. Cripps, P. B. Kenington, Z. B. Popović, N. Pothecary *et al.*, "Power amplifiers and transmitters for RF and microwave," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 3, pp. 814–826, Mar. 2002.

[219] A. Rabany, L. Nguyen, and D. Rice, "Memory effect reduction for LDMOS bias circuits," *Microwave Journal*, vol. 46, no. 2, Feb. 2003.

[220] R. Raich, H. Qian, and G. T. Zhou, "Digital baseband predistortion of nonlinear power amplifiers using orthogonal polynomials," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'03*, vol. 6, Apr. 2003, pp. 689–692.

[221] ——, "Orthogonal polynomials for power amplifier modeling and predistorter design," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 5, Sep. 2004.

[222] R. Raich and G. T. Zhou, "On the modeling of memory nonlinear effects of power amplifiers for communication applications," in *10th IEEE Digital Signal Processing Workshop*, Oct. 2002, pp. 7–10.

[223] B. Ram, *Numerical Methods*. New Jersey: Pearson, 2010.

[224] H. Ratschek and J. Rokne, *Computer Methods for the Range of Functions*, ser. Mathematics and its Applications. Halsted Press, 1984.

[225] C. Rauscher, *Fundamentals of Spectrum Analysis*. Munich: Rohde & Schwarz, 2004.

[226] Q. Ren and I. Wolff, "Effect of demodulator errors on predistortion linearization," *IEEE Transactions on Broadcasting*, vol. 45, no. 2, pp. 153–161, Jun. 1999.

[227] A. Richardson, *WCDMA Design Handbook*. New York: Cambridge University Press, 2005.

[228] P. Roblin, S. K. Myoung, D. Chaillot, Y. G. Kim, A. Fathimulla, J. Strahler, and S. Bibyk, "Frequency-selective predistortion linearization of RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 1, pp. 65–76, Jan. 2008.

[229] *AMIQ I/Q Modulation Generator Operating Manual*, 1110.2003.02/03/04, Rohde & Schwarz, Munich, 1999. [Online]. Available: www.rohde-schwarz.com

[230] *I/Q Modulation Generator R&S AMIQ: New approaches in the generation of complex I/Q signals*, PD 0757.3970.24, Rohde & Schwarz, Munich, 1999. [Online]. Available: www.rohde-schwarz.com

[231] *FSIQ26 Signal Analyzer Operating Manual*, 1119.6001.27, Rohde & Schwarz, Munich, 2000. [Online]. Available: www.rohde-schwarz.com

[232] *SMIQ06B Vector Signal Generator Operating Manual (Volumes 1-2)*, 1125.5610.12-11, Rohde & Schwarz, Munich, 2000. [Online]. Available: www.rohde-schwarz.com

[233] *Vector Signal Generator R&S SMIQ: Digital signals of your choice*, PD 0757.2438.27, Rohde & Schwarz, Munich, 2000. [Online]. Available: www.rohde-schwarz.com

[234] "The crest factor in DVB-T (OFDM) transmitter systems and its influence on the dimensioning of power components," Application Note 7TS02, Rohde & Schwarz, Munich, January 2007. [Online]. Available: www.rohde-schwarz.com

[235] *FSL Spectrum Analyzer Operating Manual*, 1300.2519.12-12, Rohde & Schwarz, Munich, 2008. [Online]. Available: www.rohde-schwarz.com

[236] "R&S NH/NV8600 UHF transmitter family for TV," Data Sheet, Rohde & Schwarz, Munich, January 2011. [Online]. Available: www.rohde-schwarz.com

[237] *Band IV/V Medium Power Amplifier Instrument Manual*, R&S VH60xxA2, Rohde & Schwarz Broadcasting Division, Munich, Sep. 2006.

[238] D. Rönnow, "Applying digital predistortion to RF/microwave PAs," *Wireless Design Magazine*, pp. 38–43, Jun. 2004. [Online]. Available: www.wirelessdesignmag.com

[239] D. Rönnow and M. Isaksson, "Digital predistortion of radio frequency power amplifiers using Kautz-Volterra model," *Electronics Letters*, vol. 42, no. 13, pp. 780–782, Jun. 2006.

[240] W. J. Rugh, *Nonlinear System Theory: The Volterra / Wiener Approach.* Maryland: The John Hopkins University Press, 1981.

[241] N. Safari, T. Røste, P. Fedorenko, and J. Stevenson Kenney, "An approximation of Volterra series using delay envelopes, applied to digital predistortion of RF power amplifiers with memory effects," *IEEE Microwave and Wireless Components Letters*, vol. 18, no. 2, pp. 115–117, Feb. 2008.

[242] A. A. M. Saleh and J. Salz, "Adaptive linearization of power amplifiers in digital radio systems," *Bell Systems Technical Journal*, vol. 62, no. 4, pp. 1019–1033, Apr. 1983.

[243] A. Sano and L. Sun, "Identification of Hammerstein-Wiener system with application to compensation for nonlinear distortion," in *41st SICE Annual Conference*, vol. 3, Aug. 2002, pp. 1521–1526.

[244] I. Santamaría, J. Ibáñez, M. Lázaro, C. Pantaleón, and L. Vielva, "Modeling nonlinear power amplifiers in OFDM systems from subsampled data: a comparative study using real measurements," *EURASIP Journal on Applied Signal Processing*, no. 12, pp. 1219–1228, Dec. 2003.

[245] V. Sarreiter, "Liquid-cooled TV transmitters for terrestrial digital TV," *News From Rohde & Schwarz*, vol. 39, no. 165, pp. 11–13, 1999. [Online]. Available: www.rohde-schwarz.com

[246] Y. Sawaragi, H. Nakayama, and T. Tanino, *Theory of Multiobjective Optimization*. Orlando: Academic Press, 1985, vol. 176 Mathematics in Science and Engineering.

[247] M. Schetzen, "Theory of pth-order inverses of nonlinear systems," *IEEE Transactions on Circuits and Systems*, vol. 23, no. 5, pp. 285–291, May 1976.

[248] ——, *The Volterra And Wiener Theories Of Nonlinear Systems*, 3rd ed. Florida: Kreiger Publishing Company, 2006.

[249] H. Schildt, *C++: The Complete Reference*, 4th ed. California: McGraw-Hill/Osborne, 2003.

[250] P. B. Seel, *Digital Universe: The Global Telecommunication Revolution*. Oxford: Wiley-Blackwell, 2012.

[251] H. Seidel, H. Beurrier, and A. Friedman, "Error controlled high power linear amplifiers at VHF," *Bell Systems Technical Journal*, vol. 47, pp. 651–722, May/Jun. 1968.

[252] J. F. Sevic and M. B. Steer, "On the significance of envelope peak-to-average ratio for estimating the spectral regrowth of an RF/Microwave power amplifier," *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, no. 6, pp. 1068–1071, Jun. 2000.

[253] W. Shan and L. Sundström, "Effects of anti-aliasing filters in feedback path of adaptive predistortion," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 1, 2002, pp. 469–472.

[254] K. S. Shanmugan and M. J. Ruggles, "An adaptive linearizer for 16 QAM transmission over nonlinear satellite channels," in *Global Telecommunications Conference, GLOBECOM'86*, Dec. 1986, pp. 226–230.

[255] C. P. Silva, A. A. Moulthrop, and M. S. Muha, "Introduction to polyspectral modeling and compensation techniques for wideband communications systems," in *58th ARFTG Conference Digest-Fall*, vol. 40, Nov. 2001, pp. 1–15.

[256] ——, "Polyspectral techniques for nonlinear system modeling and distortion compensation," in *IEEE 3rd International Vacuum Electronics Conference, IVEC'02*, 2002, pp. 314–315.

[257] C. P. Silva, A. A. Moulthrop, M. S. Muha, and C. J. Clark, "Application of polyspectral techniques to nonlinear modeling and compensation," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 1, 2001, pp. 13–16.

[258] D. Silveira, M. Gadringer, H. Arthaber, and G. Magerl, "RF-power amplifier characteristics determination using parallel cascade Wiener models and pseudo-inverse techniques," in *Asia-Pacific Microwave Conference, APMC'05*, vol. 1, Dec. 2005, p. 4.

[259] A. Skinner, *The Digital Evolution of Broadcast Technology.* Peterborough: FastPrint, Apr. 2010.

[260] W. Spendley, G. R. Hext, and F. R. Himsworth, "Sequential application of simplex designs in optimisation and evolutionary operation," *Technometrics*, vol. 4, p. 441, 1962.

[261] "Broadcast components and systems," Product Catalogue, Spinner GMBH, Munich, pp. 36–55, 2006. [Online]. Available: www.spinner.de

[262] S. P. Stapleton and J. K. Cavers, "A new technique for adaption of linearizing predistorters," in *41st IEEE Vehicular Technology Conference*, May 1991, pp. 753–758.

[263] S. P. Stapleton and F. C. Costescu, "An adaptive predistorter for a power amplifier based on adjacent channel emissions," *IEEE Transactions on Vehicular Technology*, vol. 41, no. 1, pp. 49–56, Feb. 1992.

[264] ——, "An adaptive predistortion system," in *42nd IEEE Vehicular Technology Conference*, vol. 2, May 1992, pp. 690–693.

[265] S. P. Stapleton, G. S. Kandola, and J. K. Cavers, "Simulation and analysis of an adaptive predistorter utilizing a complex spectral convolution," *IEEE Transactions on Vehicular Technology*, vol. 41, no. 4, pp. 387–394, Nov. 1992.

[266] R. E. Steuer, *Multiple Criteria Optimization: Theory, Computations and Application.* New York: John Wiley & Sons, 1986.

[267] B. Stroustrup, *The C++ Programming Language*, 2nd ed. Boston: Addison-Wesley, 2008.

[268] L. Sundström, M. Faulkner, and M. Johansson, "Effects of reconstruction filters in digital predistortion linearizers for RF power amplifiers," *IEEE Transactions on Vehicular Technology*, vol. 44, no. 1, pp. 131–139, Feb. 1995.

[269] ——, "Quantization analysis and design of a digital predistortion linearizer for RF power amplifiers," *IEEE Transactions on Vehicular Technology*, vol. 45, no. 4, pp. 707–719, Nov. 1996.

[270] I. Teikari and K. Halonen, "The effect of quadrature modulator nonlinearity on a digital baseband predistortion system," in *9th European Conference on Wireless Technology*, Sep. 2006, pp. 342–345.

[271] D. Torrieri, *Principles of Spread-Spectrum Communication Systems.* Boston: Springer, 2005.

[272] J. Tsimbinos and K. V. Lever, "Computational complexity of Volterra based nonlinear compensators," *Electronics Letters*, vol. 32, no. 9, pp. 852–854, Apr. 1996.

[273] R. C. Tupynambá and E. Camargo, "MESFET nonlinearities applied to predistortion linearizer design," *Microwave Symposium Digest, IEEE MTT-S International*, vol. 2, pp. 955–958, Jun. 1992.

[274] T. R. Turlington, Ed., *Behavioral Modeling of Nonlinear RF and Microwave Devices*, ser. Microwave Library. Massachusetts: Artech House, 2000.

[275] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation and Modulation Theory.* New York: John Wiley & Sons, 2002.

[276] V. Volterra, "Sopra le funzioni che dipendono de altre funzioni," *Rend. R. Accademia dei Lincei 2°Sem*, pp. 97–105, 1887.

[277] ——, *Lecons sur les Fonctions De Lignes.* Paris: Gauthier-Villars, 1913.

[278] J. Vuolevi and T. Rahkonen, *Distortion in RF Power Amplifiers.* Massachusetts: Artech House, 2003.

[279] J. H. K. Vuolevi, T. Rahkonen, and J. P. A. Manninen, "Measurement technique for characterizing memory effects in RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 8, pp. 1383–1389, Aug. 2001.

[280] P. Wambacq and W. Sansen, *Distortion Analysis of Analog Integrated Circuits*, ser. Analog Circuits and Signal Processing, M. Ismail, Ed. Dordrecht: Kluwer Academic, 1998.

[281] T. Wang and J. Ilow, "Compensation of nonlinear distortions with memory effects in digital transmitters," in *Communication Networks and Services Research Conference*, no. 1, May 2004, pp. 3–9.

[282] B. E. Watkins, R. North, and M. Tummala, "Neural network based adaptive predistortion for the linearization of nonlinear RF amplifiers," in *IEEE Military Communications Conference, MILCOM'95*, no. 1, Nov. 1995, pp. 145–149.

[283] T. Weise. (2009, June) Global optimization algorithms - theory and application. Ebook. University of Kassel, Distributed Systems Group. [Online]. Available: www.it-weise.de

[284] L. D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, June 1994.

[285] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, ser. Signal Processing. New Jersey: Prentice-Hall, 1985.

[286] J. Wolf, "Measurement of adjacent channel power on wideband CDMA signals," Application Note 1EF40-OE, Rohde & Schwarz, Munich, March 1998. [Online]. Available: www.rohde-schwarz.com

[287] J. Wolf and K. Tiepermann, "Measuring ACPR of W-CDMA signals with a spectrum analyzer," Digital Modulation Technical Series, Tektronix, September 1998. [Online]. Available: www.tek.com

[288] Y. Y. Woo, J. Kim, J. Yi, S. Hong, I. Kim, J. Moon, and B. Kim, "Adaptive digital feedback predistortion technique for linearizing power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 55, no. 5, pp. 932–940, May 2007.

[289] J. Wood and D. E. Root, Eds., *Fundamentals of Nonlinear Behavioral Modeling For RF and Microwave Design.* Massachusetts: Artech House, 2005.

[290] A. S. Wright and W. G. Durtler, "Experimental performance of an adaptive digital linearized power amplifier," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 2, Jun. 1992, pp. 1105–1108.

[291] J. W. Wustenberg, H. J. Xing, and J. R. Cruz, "Complex gain and fixed-point digital predistorters for CDMA power amplifiers," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 2, pp. 469–478, Mar. 2004.

[292] W. Yong, X. Xin, and Y. Ke-Chu, "Baseband Wiener predistorter for linearizing power amplifier," in *International Conference on Signal Processing, ICSP'06*, vol. 3, 2006, p. 4.

[293] T. Ytterdal, Y. Cheng, and T. A. Fjeldly, *Device Modeling for Analog and RF CMOS Circuit Design*. Chichester: John Wiley & Sons, 2003.

[294] J. Zhai, J. Zhou, L. Zhang, J. Zhao, and W. Hong, "The dynamic behavioral model of RF power amplifiers with the modified ANFIS," *IEEE Transactions on Microwave Theory and Techniques*, vol. 57, no. 1, pp. 27–35, Jan. 2009.

[295] D. Zhou and V. DeBrunner, "A novel adaptive nonlinear predistorter based on the direct learning algorithm," in *IEEE International Conference on Communications*, vol. 4, Jun. 2004, pp. 2362–2366.

[296] A. Zhu, University College Dublin, Ireland, Dec. 2008, private email communication.

[297] A. Zhu and T. J. Brazil, "Optimal digital Volterra predistorter for broadband RF power amplifier linearization," in *31st European Microwave Conference*, Sep. 2001, p. 4.

[298] ——, "An adaptive Volterra predistorter for the linearization of RF power amplifiers," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 1, 2002, pp. 461–464.

[299] ——, "Behavioral modeling of RF power amplifiers based on pruned Volterra series," *IEEE Microwave and Wireless Components Letters*, vol. 14, no. 12, pp. 563–565, Dec. 2004.

[300] ——, "RF power amplifier behavioral modeling using Volterra expansion with Laguerre functions," in *Microwave Symposium Digest, IEEE MTT-S International*, Jun. 2005, p. 4.

[301] ——, "An overview of Volterra series based behavioral modeling of RF/Microwave power amplifiers," in *IEEE Wireless and Microwave Technology Conference, WAMICON'06*, Dec. 2006, pp. 1–5.

[302] A. Zhu, J. Dooley, and T. J. Brazil, "Simplified Volterra series based behavioral modeling of RF power amplifiers using deviation-reduction," in *Microwave Symposium Digest, IEEE MTT-S International*, Jun. 2006, pp. 1113–1116.

[303] A. Zhu, P. J. Draxler, J. J. Yan, T. J. Brazil, D. F. Kimball, and P. M. Asbeck, "Open-loop digital predistorter for RF power amplifiers using dynamic deviation reduction-based Volterra series," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 7, pp. 1524–1534, Jul. 2008.

[304] A. Zhu, J. C. Pedro, and T. J. Brazil, "Dynamic deviation reduction-based Volterra behavioral modeling of RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, no. 12, pp. 4323–4332, Dec. 2006.

[305] A. Zhu, J. C. Pedro, and T. R. Cunha, "Pruning the Volterra series for behavioral modeling of power amplifiers using physical knowledge," *IEEE Transactions on Microwave Theory and Techniques*, vol. 55, no. 5, pp. 813–821, May 2007.

[306] A. Zhu, M. Wren, and T. J. Brazil, "An efficient Volterra-based behavioral model for wideband RF power amplifiers," in *Microwave Symposium Digest, IEEE MTT-S International*, vol. 2, Jun. 2003, pp. 787–790.

[307] R. E. Ziemer, W. H. Tranter, and D. R. Fannin, *Signals & Systems: Continuous and Discrete*, 4th ed. New Jersey: Prentice Hall, 1998.

[308] A. I. Zverev, *Handbook of Filter Synthesis*. New Jersey: John Wiley & Sons, 2005.

Appendix A

Shortlisted Optimization Algorithms

A.1 Gradient Descent

The Gradient Descent (GD) optimization algorithm is classified as a *line search* algorithm [179]. At each iteration $\boldsymbol{h}_k$, a general *line search* algorithm will:

1. choose a line (a.k.a. direction) along which to step
2. search along this line for a suitable step distance

For the specific Gradient Descent algorithm, the line along which to step is chosen as the line of maximum negative rate of change (steepest descent). From the theory of directional derivatives [149], this line is represented mathematically as the unit vector:

$$\boldsymbol{u} = \frac{-\boldsymbol{g}}{\|\boldsymbol{g}\|} \tag{A.1}$$

where the Gradient vector $\boldsymbol{g}$ is defined as:

$$\boldsymbol{g} \triangleq \frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}}\bigg|_{\boldsymbol{h}_k} = \begin{bmatrix} \frac{\partial B(\boldsymbol{h})}{\partial h_1} & \frac{\partial B(\boldsymbol{h})}{\partial h_2} & \cdots & \frac{\partial B(\boldsymbol{h})}{\partial h_n} \end{bmatrix}^T \bigg|_{\boldsymbol{h}_k} \tag{A.2}$$

When $B(\boldsymbol{h})$ has no closed form, $\boldsymbol{g}$ must be estimated numerically via the *Finite Differences* or *Least Squares* techniques outlined in Appendix B.1.

To reap the maximum reward from this line of step $\boldsymbol{u}$, the ideal step distance μ^{ideal} is computed as:

$$\mu^{\text{ideal}} = \arg\min_{l>0} B(\boldsymbol{h}_k + l\boldsymbol{u}) \tag{A.3}$$

Solving (A.3) is no trivial task however, generally requiring an excessive number of objective function measurements and gradient estimates. Since computational

complexity is undesirable in any optimization algorithm, practical implementations of the Gradient Descent algorithm generally trade the ideal step of (A.3) for a nonideal step μ requiring fewer computations.

While many computationally efficient methods exist for generating a μ value, they all share the common trait of not computing μ directly, instead selecting μ from a set of candidate steps [200]. The simple but reliable method used in this research's implementation of the Gradient Descent algorithm is outlined as follows:

- Let $\mu_{\boldsymbol{h}_k}$ represent the step distance chosen for iteration $\boldsymbol{h}_k$ and $c^i_{\boldsymbol{h}_k}$ represent the i^{th} candidate step distance generated during iteration $\boldsymbol{h}_k$.

- For iteration $\boldsymbol{h}_k$, candidate steps are generated as:

$$c^i_{\boldsymbol{h}_k} = \begin{cases} 1.05\,\mu_{\boldsymbol{h}_{k-1}} & i = 1 \\ 0.95\,c^{i-1}_{\boldsymbol{h}_k} & i \geq 2 \end{cases} \tag{A.4}$$

  Starting from $c^1_{\boldsymbol{h}_k}$, the first candidate step that brings about a reduction in the objective is chosen as $\mu_{\boldsymbol{h}_k}$. It is noted that for the very first iteration $\boldsymbol{h}_1$, no previous step $\mu_{\boldsymbol{h}_0}$ exists for use in (A.4), in which case $\mu_{\boldsymbol{h}_0}$ is chosen based on *a priori* knowledge.

- Four points are worth noting about this method for generating a step distance:

  1. Since $c^1_{\boldsymbol{h}_k} > \mu_{\boldsymbol{h}_{k-1}}$, the step length is capable of growing.
  2. Since $c^i_{\boldsymbol{h}_k} < \mu_{\boldsymbol{h}_{k-1}}$ for $i \geq 2$, the step length is capable of shrinking.
  3. Since $c^i_{\boldsymbol{h}_k} < c^{i-1}_{\boldsymbol{h}_k}$, the method is guaranteed to identify a step length which reduces the objective function and is hence considered reliable.
  4. Since $\mu_{\boldsymbol{h}_k}$ is chosen as the first candidate step bringing about a reduction in the objective function, subsequent better candidate steps could potentially be missed. This is quite simply the tradeoff accepted for the method's computational efficiency.

A theoretically automated exit point from the Gradient Descent algorithm cannot be defined based on first order objective function characteristics alone. While a local minimum exhibits $\|\boldsymbol{g}\| = 0$, so too does an objective function maximum or saddle point. If at a particular iteration $\|\boldsymbol{g}\| = 0$, differentiation between the three possible cases requires computation of second order objective function characteristics in the form of a Hessian matrix. A local minimum is subsequently identified if the Hessian matrix is positive-definite. This being said however, if the objective function has no closed form, estimation of this Hessian matrix is computationally intensive and therefore generally avoided. In such cases, exiting from the Gradient Descent algorithm is generally left to the user's discretion, for example, when

the algorithm continually fails to provide a meaningful objective function reduction.

This Gradient Descent algorithm is summarized in the flowchart of Figure A.1. It is also implemented on the laboratory testbed via the *GradientDescent()* function. Corresponding declaration and definition source code resides in project files *GradientDescentOptimization.h* and *GradientDescentOptimization_Templates.cpp* respectively; both files located within folder *Software\Cpp* on the accompanying DVD.

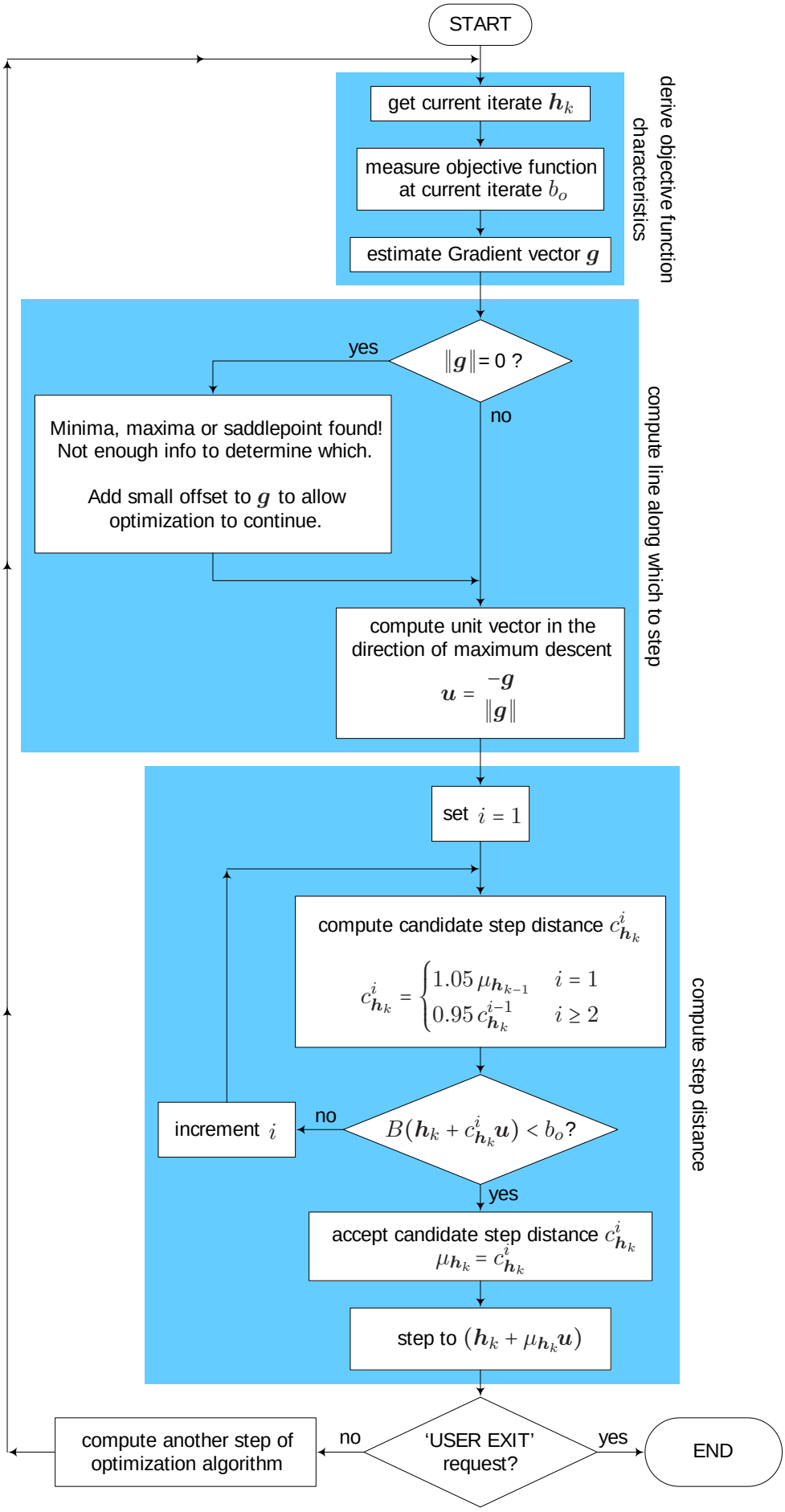


Figure A.1: Flowchart of Gradient Descent optimization algorithm

A.2 Trust Region Newton

At each iteration $\boldsymbol{h}_k$ of the Trust Region Newton (TRN) optimization algorithm, the objective function is approximated by the second order Taylor series:

$$B(\,\boldsymbol{h}_k + \boldsymbol{d}\,) \;\approx\; B(\boldsymbol{h}_k) + \boldsymbol{d}^T \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \bigg|_{\boldsymbol{h}_k} \right) + \frac{1}{2} \boldsymbol{d}^T \left(\frac{\partial}{\partial \boldsymbol{h}} \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \bigg|_{\boldsymbol{h}_k} \right) \boldsymbol{d} \tag{A.5}$$

The right hand side of (A.5) thus represents a quadratic model of the objective function at the current iteration $\boldsymbol{h}_k$:

$$m(\,\boldsymbol{d}\,) \;\triangleq\; B(\boldsymbol{h}_k) + \boldsymbol{d}^T \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \bigg|_{\boldsymbol{h}_k} \right) + \frac{1}{2} \boldsymbol{d}^T \left(\frac{\partial}{\partial \boldsymbol{h}} \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \bigg|_{\boldsymbol{h}_k} \right) \boldsymbol{d} \tag{A.6}$$

It can be seen from (A.6) that at the current iterate, the model and objective function characteristics are identical up to second order. That is:

$$m(\,\boldsymbol{d}\,)\big|_{\boldsymbol{d}=\boldsymbol{0}} = \; B(\boldsymbol{h}_k) \tag{A.7}$$

$$\frac{\partial\, m(\,\boldsymbol{d}\,)}{\partial \boldsymbol{d}} \bigg|_{\boldsymbol{d}=\boldsymbol{0}} = \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \bigg|_{\boldsymbol{h}_k} \right) + \left(\frac{\partial}{\partial \boldsymbol{h}} \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \bigg|_{\boldsymbol{h}_k} \right) \boldsymbol{d} \;\bigg|_{\boldsymbol{d}=\boldsymbol{0}} = \; \frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \bigg|_{\boldsymbol{h}_k} \tag{A.8}$$

$$\frac{\partial}{\partial \boldsymbol{d}} \left(\frac{\partial m(\,\boldsymbol{d}\,)}{\partial \boldsymbol{d}} \right)^T \bigg|_{\boldsymbol{d}=\boldsymbol{0}} = \left(\frac{\partial}{\partial \boldsymbol{h}} \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \bigg|_{\boldsymbol{h}_k} \right) \bigg|_{\boldsymbol{d}=\boldsymbol{0}} = \; \frac{\partial}{\partial \boldsymbol{h}} \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \bigg|_{\boldsymbol{h}_k} \tag{A.9}$$

To simplify notation, the (A.6) model can be rewritten as:

$$m(\,\boldsymbol{d}\,) \;\triangleq\; b_o + \boldsymbol{d}^T \boldsymbol{g} + \frac{1}{2} \boldsymbol{d}^T \boldsymbol{H} \boldsymbol{d} \tag{A.10}$$

where

$$b_o \triangleq B(\boldsymbol{h}_k) \tag{A.11}$$

$$\boldsymbol{g} \triangleq \left.\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}}\right|_{\boldsymbol{h}_k} = \left.\begin{bmatrix} \frac{\partial B(\boldsymbol{h})}{\partial h_1} & \frac{\partial B(\boldsymbol{h})}{\partial h_2} & \cdots & \frac{\partial B(\boldsymbol{h})}{\partial h_n} \end{bmatrix}^T\right|_{\boldsymbol{h}_k} \tag{A.12}$$

$$\boldsymbol{H} \triangleq \left.\frac{\partial}{\partial \boldsymbol{h}}\left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}}\right)^T\right|_{\boldsymbol{h}_k} = \left.\begin{bmatrix} \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1\,\partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1\,\partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1\,\partial h_n} \\ \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2\,\partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2\,\partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2\,\partial h_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n\,\partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n\,\partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n\,\partial h_n} \end{bmatrix}\right|_{\boldsymbol{h}_k} \tag{A.13}$$

$\boldsymbol{g}$ and $\boldsymbol{H}$ are referred to as the Gradient vector and Hessian matrix respectively. When the objective function $B(\boldsymbol{h})$ has no closed form, $\boldsymbol{g}$ and $\boldsymbol{H}$ must be estimated numerically via the techniques of Appendix B.1 and B.2 respectively. $\boldsymbol{H}$ may also be estimated via *Symmetric-Rank-1 Updating*, as discussed later in this section, for all iterations other than the very first. In all such cases where $\boldsymbol{g}$ and $\boldsymbol{H}$ are estimated numerically, the optimization algorithm is technically referred to as the Trust Region *Quasi*-Newton algorithm. It is noted that both $\boldsymbol{g}$ and $\boldsymbol{H}$ are real and $\boldsymbol{H}$ is symmetric.

In order to quantify the domain of this model $m(\boldsymbol{d})$, an accompanying *trust region radius* $\Delta > 0$ must be specified. A *trust region* is defined as a spherical region (ball of radius Δ) around the current iterate within which the quadratic model $m(\boldsymbol{d})$ is *trusted* to be an accurate representation of the objective function $B(\boldsymbol{h}_k + \boldsymbol{d})$.

When both the quadratic model $m(\boldsymbol{d})$ and its accompanying *trust region radius* Δ are estimated at the current iteration $\boldsymbol{h}_k$, a candidate optimization step is computed by minimizing the model within the current *trust region*:

$$\min_{\boldsymbol{d}} m(\boldsymbol{d}) = b_o + \boldsymbol{d}^T\boldsymbol{g} + \frac{1}{2}\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d} \qquad \text{subject to} \qquad \|\boldsymbol{d}\| \le \Delta \tag{A.14}$$

This minimizer $\boldsymbol{d}^*$ represents the candidate optimization step. If $m(\boldsymbol{d}^*)$ and $B(\boldsymbol{h}_k + \boldsymbol{d}^*)$ are comparable, the candidate step is locked in. If however $m(\boldsymbol{d}^*)$ and $B(\boldsymbol{h}_k + \boldsymbol{d}^*)$

are significantly different, the current Δ estimate is considered to be over-estimated. In this case, the Δ estimate must be reduced and the minimization of (A.14) re-computed. In general, the candidate optimization step will be different for each Δ estimate.

The size of the *trust region radius* estimate is crucial in the effectiveness of this algorithm. If the radius is under-estimated, the algorithm will miss the opportunity to take a larger step towards the objective function minimum and the algorithm will take longer to converge. If on the other hand the radius is over-estimated, the model won't adequately represent the objective function over the entire region and the candidate step will in general be misleading. In practice, the initial *trust region radius* estimate at iteration $\boldsymbol{h}_k$ is chosen to be slightly larger than the final estimate of the previous iteration $\boldsymbol{h}_{k-1}$. This allows the trust region radius to grow, thus avoiding potential future under-estimation and slow convergence, at the same time keeping the number of reduction re-estimates at each iteration to a minimum.

[185] states that the solution $\boldsymbol{d}^*$ of (A.14) must satisfy the following conditions for some scalar $\lambda \geq 0$:

$$[\boldsymbol{H} + \lambda \boldsymbol{I}]\, \boldsymbol{d}^* = -\boldsymbol{g} \tag{A.15a}$$

$$\lambda\,(\Delta - \|\boldsymbol{d}^*\|) = 0 \tag{A.15b}$$

$$\boldsymbol{H} + \lambda \boldsymbol{I} \quad \text{is positive semidefinite} \tag{A.15c}$$

These conditions suggest a two case strategy for finding the solution $\boldsymbol{d}^*$ [200]:

Case 1: If $\lambda = 0$ satisfies (A.15c) (specifically $\boldsymbol{H}$ is positive-definite and therefore nonsingular) and the solution $\boldsymbol{d}^* = -\boldsymbol{H}^{-1}\boldsymbol{g}$ of (A.15a) is within the trust region ($\|\boldsymbol{d}^*\| \leq \Delta$) then $\boldsymbol{d}^*$ represents the solution of (A.14). This solution is known as the *unconstrained minimum* of the model $m(\boldsymbol{d})$.

Case 2: If Case 1 does not hold, the solution $\boldsymbol{d}^*$ of (A.15a) must be evaluated as a function of λ:

$$\boldsymbol{d}^*(\lambda) = -\,[\boldsymbol{H} + \lambda \boldsymbol{I}]^{-1}\, \boldsymbol{g} \tag{A.16}$$

specifically for λ satisfying (A.15c) and the solution of (A.14) must be chosen as that $\boldsymbol{d}^*(\lambda)$ for which $\|\boldsymbol{d}^*(\lambda)\| = \Delta$, thus satisfying (A.15b).

Before continuing, the reader is encouraged to consult Appendix B.3 to gain familiarity with the eigen properties of $\boldsymbol{H}$. Knowledge of these properties will be assumed in the following discussion.

Evaluating The Solution Via Case 1

The first step in evaluating the solution of (A.14) via Case 1 is to determine whether $\boldsymbol{H}$ is positive-definite or not. $\boldsymbol{H}$ is positive-definite if all of its eigenvalues are positive (Property 4, Appendix B.3). The eigenvalues of $\boldsymbol{H}$ are obtained by decomposing $\boldsymbol{H}$ into its diagonal form $\boldsymbol{H} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$ via the QR-Factorization method [88] and stripping off the diagonal elements of the spectral matrix $\boldsymbol{\Lambda}$ (Property 2, Appendix B.3).

If these eigenvalues are all positive and $\boldsymbol{H}$ is subsequently positive-definite, the *unconstrained minimum solution* $\boldsymbol{d}^* = -\boldsymbol{H}^{-1}\boldsymbol{g}$ of (A.15a) must be evaluated. Rather than computing a matrix inverse however, $\boldsymbol{H}\boldsymbol{d}^* = -\boldsymbol{g}$ is treated as a linear system of equations and solved efficiently via Cholesky decomposition [143].

Once the above *unconstrained minimum* $\boldsymbol{d}^*$ has been computed, its position with respect to the trust region must be determined. If:

$$\|\boldsymbol{d}^*\| = \left(\boldsymbol{d}^{*T}\boldsymbol{d}^*\right)^{\frac{1}{2}} \leq \Delta \tag{A.17}$$

the *unconstrained minimum* lies within the trust region and therefore represents the solution of (A.14) via Case 1.

Evaluating The Solution Via Case 2

If $\boldsymbol{H}$ is not positive-definite or the *unconstrained minimum* is not within the trust region, the solution of (A.14) must be evaluated via Case 2. In this case, the solution $\boldsymbol{d}^*$ of (A.15a) must be evaluated as a function of λ:

$$\boldsymbol{d}^*(\lambda) = -\left[\boldsymbol{H} + \lambda\boldsymbol{I}\right]^{-1}\boldsymbol{g} \tag{A.18}$$

specifically for λ satisfying (A.15c) and the solution of (A.14) must be chosen as that $\boldsymbol{d}^*(\lambda)$ for which $\|\boldsymbol{d}^*(\lambda)\| = \Delta$, thus satisfying (A.15b).

Decomposing $[\boldsymbol{H} + \lambda\boldsymbol{I}]$ into its diagonal form $\boldsymbol{Q}[\boldsymbol{\Lambda} + \lambda\boldsymbol{I}]\boldsymbol{Q}^T$ (Property 2 & 5, Appendix B.3) and substituting into (A.18) gives:

$$\boldsymbol{d}^*(\lambda) = -\boldsymbol{Q}\left[\boldsymbol{\Lambda} + \lambda\boldsymbol{I}\right]^{-1}\boldsymbol{Q}^T\boldsymbol{g} \tag{A.19}$$

$$= -\sum_{i=1}^{n}\left(\frac{\boldsymbol{q}_i^T\boldsymbol{g}}{\lambda_i + \lambda}\right)\boldsymbol{q}_i \tag{A.20}$$

where $\boldsymbol{q}_i$ represents the i^{th} eigenvector of $\boldsymbol{H}$ and therefore the i^{th} column of $\boldsymbol{Q}$. Given the orthogonality of $\boldsymbol{Q}$, $\|\boldsymbol{d}^*(\lambda)\|$ can be expressed as:

$$\|\boldsymbol{d}^*(\lambda)\| = \left(\sum_{i=1}^{n}\left(\frac{\boldsymbol{q}_i^T\boldsymbol{g}}{\lambda_i + \lambda}\right)^2\right)^{\frac{1}{2}} \tag{A.21}$$

As stated above, the goal now is to find that value of λ satisfying (A.15c), for which $\|\boldsymbol{d}^*(\lambda)\| = \Delta$. The corresponding $\boldsymbol{d}^*(\lambda)$ subsequently represents the solution of (A.14). From (A.21), it can be seen that two scenarios exist in this search for λ.

Case 2 - Scenario 1 When $\boldsymbol{q}_1^T\boldsymbol{g} \neq 0$, $\|\boldsymbol{d}^*(\lambda)\|$ in (A.21) is a continuous, nonincreasing function of $\lambda \in (-\lambda_1, \infty)$ for which:

$$\lim_{\lambda \to -\lambda_1} \|\boldsymbol{d}^*(\lambda)\| = \infty \quad \text{and} \quad \lim_{\lambda \to \infty} \|\boldsymbol{d}^*(\lambda)\| = 0 \tag{A.22}$$

In this scenario, a unique $\lambda \in (-\lambda_1, \infty)$ exists for which $\|\boldsymbol{d}^*(\lambda)\| = \Delta$. This λ is computed via Newton's root-finding method [200] as follows.

On the open interval $\lambda \in (-\lambda_1, \infty)$, the function:

$$\phi(\lambda) = \|\boldsymbol{d}^*(\lambda)\| - \Delta \tag{A.23}$$

will have a root at the unique value of λ for which $\|\boldsymbol{d}^*(\lambda)\| = \Delta$. Given an initial estimate of the root $\lambda^0 \in (-\lambda_1, \infty)$, the iterative Newton's root-finding method estimates an improved estimate according to:

$$\lambda^{k+1} = \lambda^k - \frac{\phi(\lambda^k)}{\phi'(\lambda^k)} \quad \text{where} \quad \phi'(\lambda^k) = \left.\frac{d\phi}{d\lambda}\right|_{\lambda^k} \tag{A.24}$$

This iteration continues until $\phi(\lambda^{k+1})$ approaches zero at which point λ^{k+1} represents the desired value of λ and the corresponding $\boldsymbol{d}^*(\lambda)$ represents the solution of (A.14). In practice, safeguards are put in place to ensure λ^{k+1} remains greater than $-\lambda_1$ during the iteration process.

The performance of the Newton's root-finding method improves with the linearity of the function $\phi(\lambda)$. For this reason, the alternative, more linear function:

$$\phi(\lambda) = \frac{1}{\Delta} - \frac{1}{\|\boldsymbol{d}^*(\lambda)\|} \tag{A.25}$$

is preferred over (A.23) which is significantly nonlinear when λ is greater than but close to $-\lambda_1$. If (A.25) is indeed utilized, [185] shows that (A.24) can be computed efficiently as:

$$\lambda^{k+1} = \lambda^k + \left(\frac{\|\boldsymbol{d}^k\|}{\|\boldsymbol{q}^k\|}\right)^2 \left(\frac{\|\boldsymbol{d}^k\| - \Delta}{\Delta}\right) \tag{A.26}$$

where $\boldsymbol{d}^k$ and $\boldsymbol{q}^k$ are computed according to:

$$\boldsymbol{H} + \lambda^k \boldsymbol{I} = \boldsymbol{R}^T \boldsymbol{R} \tag{A.27a}$$

$$\boldsymbol{R}^T \boldsymbol{R}\, \boldsymbol{d}^k = -\boldsymbol{g} \tag{A.27b}$$

$$\boldsymbol{R}^T \boldsymbol{q}^k = \boldsymbol{d}^k \tag{A.27c}$$

Here, $\boldsymbol{R}$ represents the upper triangular Cholesky factorization matrix.

Case 2 - Scenario 2 When the current iteration $\boldsymbol{h}_k$ is positioned on one of the principle axes defined by eigenvectors $\boldsymbol{q}_2 \rightarrow \boldsymbol{q}_n$ and consequently $\boldsymbol{q}_1^T \boldsymbol{g} = 0$ (Property 3, Appendix B.3), $\|\boldsymbol{d}^*(\lambda)\|$ in (A.21) is once again a continuous, nonincreasing function of $\lambda \in (-\lambda_1, \infty)$ but this time:

$$\lim_{\lambda \to -\lambda_1} \|\boldsymbol{d}^*(\lambda)\| = \left(\sum_{i=2}^{n} \left(\frac{\boldsymbol{q}_i^T \boldsymbol{g}}{\lambda_i - \lambda_1} \right)^2 \right)^{\frac{1}{2}} \quad \text{and} \quad \lim_{\lambda \to \infty} \|\boldsymbol{d}^*(\lambda)\| = 0 \tag{A.28}$$

If the trust region radius Δ is such that:

$$\Delta < \lim_{\lambda \to -\lambda_1} \|\boldsymbol{d}^*(\lambda)\| \tag{A.29}$$

then a unique $\lambda \in (-\lambda_1, \infty)$ exists for which $\|\boldsymbol{d}^*(\lambda)\| = \Delta$ and the corresponding $\boldsymbol{d}^*(\lambda)$ represents the solution of (A.14). This λ is computed via Newton's root-finding method in exactly the same manner as in Scenario 1 previously.
If however the trust region radius Δ is such that:

$$\Delta \geq \lim_{\lambda \to -\lambda_1} \|\boldsymbol{d}^*(\lambda)\| \tag{A.30}$$

then a unique value $\lambda \in (-\lambda_1, \infty)$ will not exist for which $\|\boldsymbol{d}^*(\lambda)\| = \Delta$. According to (A.15c), in this rare case λ must take on the value $-\lambda_1$. For $\lambda = -\lambda_1$, $[\boldsymbol{H} - \lambda_1 \boldsymbol{I}]$ is singular (at least one eigenvalue equals zero and therefore the determinant equals zero (Property 6, Appendix B.3)) and (A.15a) has an infinite number of solutions of the form:

$$\boldsymbol{d}^* = \left(-\sum_{i=2}^{n} \left(\frac{\boldsymbol{q}_i^T \boldsymbol{g}}{\lambda_i - \lambda_1} \right) \boldsymbol{q}_i \right) + \tau \boldsymbol{q}_1 \quad \text{where} \quad -\infty < \tau < \infty \tag{A.31}$$

From all of these solutions, the goal is to choose that unique solution $\boldsymbol{d}^*$ for which $\|\boldsymbol{d}^*\| = \Delta$. Taking the norm of (A.31) and equating to Δ gives:

$$\|\boldsymbol{d}^*\| = \left[\left(\sum_{i=2}^{n} \left(\frac{\boldsymbol{q}_i^T \boldsymbol{g}}{\lambda_i - \lambda_1} \right)^2 \right) + \tau^2 \right]^{\frac{1}{2}} = \Delta \tag{A.32}$$

Rearranging (A.32) then produces the expression for τ:

$$\tau = \left[\Delta^2 - \left(\sum_{i=2}^{n}\left(\frac{\boldsymbol{q}_i^T \boldsymbol{g}}{\lambda_i - \lambda_1}\right)^2\right)\right]^{\frac{1}{2}} \tag{A.33}$$

corresponding to the unique solution $\boldsymbol{d}^*$ of (A.31) for which $\|\boldsymbol{d}^*\| = \Delta$. This $\boldsymbol{d}^*$ subsequently represents the solution of (A.14). To be precise, it is noted that two solutions $\boldsymbol{d}^*$ actually exist for which $\|\boldsymbol{d}^*\| = \Delta$. These solutions corresponding to τ being positive and negative in (A.33). As a standard operating procedure, the solution corresponding to positive τ is chosen. It is worth noting that [185] refers to this **Case 2 - Scenario 2** as the *hard case* and the left hand limit of (A.28) as the *limiting trust region radius*.

Estimating $\boldsymbol{H}$ Via Symmetric-Rank-1 Updating

For the very first iteration of the Trust Region *Quasi*-Newton algorithm, $\boldsymbol{H}$ must be estimated via the techniques of Appendix B.2. For all subsequent iterations however, an additional technique for estimating $\boldsymbol{H}$ becomes available, this technique being *Symmetric-Rank-1 Updating*.

Let $\boldsymbol{H}_k$ and $\boldsymbol{H}_{k-1}$ represent the Hessian matrices at iterations $\boldsymbol{h}_k$ and $\boldsymbol{h}_{k-1}$ respectively. Similarly, let $\boldsymbol{g}_k$ and $\boldsymbol{g}_{k-1}$ represent the Gradient vectors at iterations $\boldsymbol{h}_k$ and $\boldsymbol{h}_{k-1}$ respectively. Instead of estimating $\boldsymbol{H}_k$ afresh at iteration $\boldsymbol{h}_k$, the *Symmetric-Rank-1 Updating* technique estimates $\boldsymbol{H}_k$ from the already available $\boldsymbol{H}_{k-1}$, $\boldsymbol{g}_{k-1}$ and $\boldsymbol{g}_k$ estimates, making it significantly more efficient than the techniques of Appendix B.2. This efficiency, coupled with good estimation accuracy [200], makes the technique very popular in practical implementations of the Trust Region *Quasi*-Newton algorithm.

In the *Symmetric-Rank-1 Updating* technique, the estimate of $\boldsymbol{H}_k$ must satisfy two conditions [74]:

Condition 1

$$\boldsymbol{H}_k\left(\boldsymbol{h}_k - \boldsymbol{h}_{k-1}\right) = \boldsymbol{g}_k - \boldsymbol{g}_{k-1} \tag{A.34}$$

This is called the *Secant* equation and ensures that $\boldsymbol{H}_k$ is estimated such that:

$$\frac{\partial m_k(\boldsymbol{d})}{\partial \boldsymbol{d}}\bigg|_{\boldsymbol{d}=-(\boldsymbol{h}_k-\boldsymbol{h}_{k-1})} = \boldsymbol{g}_k + \boldsymbol{H}_k\boldsymbol{d}\bigg|_{\boldsymbol{d}=-(\boldsymbol{h}_k-\boldsymbol{h}_{k-1})} \equiv \boldsymbol{g}_{k-1} \tag{A.35}$$

where $m_k(\boldsymbol{d})$ is the quadratic objective function model (A.10) at iteration $\boldsymbol{h}_k$. Put simply, this condition ensures that the Gradient of $m_k(\boldsymbol{d})$ matches the Gradient of the objective function at the current and previous iteration.

Condition 2

$$\boldsymbol{H}_k = \boldsymbol{H}_{k-1} + \sigma\, \boldsymbol{v}\boldsymbol{v}^T \tag{A.36}$$

This equation represents the general form of $\boldsymbol{H}_k$ where σ (+1 or −1) and $\boldsymbol{v}$ (n element column vector) are computed such that the *Secant* equation of Condition 1 is satisfied. It is called the *Symmetric-Rank-1 Update* equation since the outer product $\boldsymbol{v}\boldsymbol{v}^T$ is symmetric with unity rank [11] and $\boldsymbol{H}_k$ is obtained by updating the previous $\boldsymbol{H}_{k-1}$. It is now obvious as to the naming origins of this Hessian estimation technique.

Substituting the *Symmetric-Rank-1 Update* equation into the *Secant* equation gives:

$$\Big(\boldsymbol{H}_{k-1} + \sigma\, \boldsymbol{v}\boldsymbol{v}^T\Big)\Big(\boldsymbol{h}_k - \boldsymbol{h}_{k-1}\Big) = \boldsymbol{g}_k - \boldsymbol{g}_{k-1} \tag{A.37}$$

To simplify notation, let:

$$\boldsymbol{s}_k = \boldsymbol{h}_k - \boldsymbol{h}_{k-1} \tag{A.38}$$

$$\boldsymbol{y}_k = \boldsymbol{g}_k - \boldsymbol{g}_{k-1} \tag{A.39}$$

(A.37) can then be rewritten compactly as:

$$\Big(\boldsymbol{H}_{k-1} + \sigma\, \boldsymbol{v}\boldsymbol{v}^T\Big)\boldsymbol{s}_k = \boldsymbol{y}_k \tag{A.40}$$

Rearranging (A.40) then gives:

$$\Big[\sigma\, \boldsymbol{v}^T\boldsymbol{s}_k\Big]\boldsymbol{v} = \boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k \tag{A.41}$$

Since the square bracketed term is a scalar, it logically follows that $\boldsymbol{v}$ must be a multiple of $(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k)$. In this context, let $\boldsymbol{v}$ take the general form:

$$\boldsymbol{v} = \varphi\big(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k\big) \quad \text{for some scalar } \varphi \tag{A.42}$$

Substituting this general form of $\boldsymbol{v}$ back into (A.41) then gives:

$$\sigma\,\varphi^2\boldsymbol{s}_k^T\big(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k\big)\big(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k\big) = \boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k \tag{A.43}$$

It is clear that (A.43) is satisfied if and only if σ and φ are chosen as:

$$\sigma = \text{sign}\Big(\ \boldsymbol{s}_k^T\big(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k\big)\ \Big) \tag{A.44}$$

$$\varphi = \pm\Big|\ \boldsymbol{s}_k^T\big(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k\big)\Big|^{-\frac{1}{2}} \tag{A.45}$$

Substituting (A.44), (A.45) and (A.42) into (A.36) then gives the desired *Symmetric-*

Rank-1 Update estimate of $\boldsymbol{H}_k$:

$$\boldsymbol{H}_k = \boldsymbol{H}_{k-1} + \frac{\big(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k\big)\big(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k\big)^T}{\boldsymbol{s}_k^T\big(\boldsymbol{y}_k - \boldsymbol{H}_{k-1}\boldsymbol{s}_k\big)} \tag{A.46}$$

where $\boldsymbol{s}_k = \boldsymbol{h}_k - \boldsymbol{h}_{k-1}$ and $\boldsymbol{y}_k = \boldsymbol{g}_k - \boldsymbol{g}_{k-1}$. As mentioned previously, since this estimate of $\boldsymbol{H}_k$ is based on the already available $\boldsymbol{H}_{k-1}$, $\boldsymbol{g}_{k-1}$ and $\boldsymbol{g}_k$ estimates, it is computationally efficient. This efficiency, coupled with good estimation accuracy, makes the technique popular in practical implementations of the Trust Region *Quasi*-Newton algorithm. In the rare case when the denominator term alone approaches 0, practical implementations simply skip the update and set $\boldsymbol{H}_k = \boldsymbol{H}_{k-1}$ with negligible negative effects being observed.

It is worth noting that other Hessian updating techniques exist, for example the *BFGS* and *DFP* methods [51], however these are only appropriate for convex objective functions and hence aren't suitable for the Trust Region Newton algorithm discussed here.

This Trust Region Newton algorithm is summarized in the flowchart of Figure A.2. It is also implemented on the laboratory transmitter testbed via the software template function *TrustRegionNewton()*. Corresponding declaration and definition source code resides in project files *TrustRegionNewtonOptimization.h* and *TrustRegionNewtonOptimization Templates.cpp* respectively. Both files are located within folder *Software\Cpp* on the accompanying DVD.

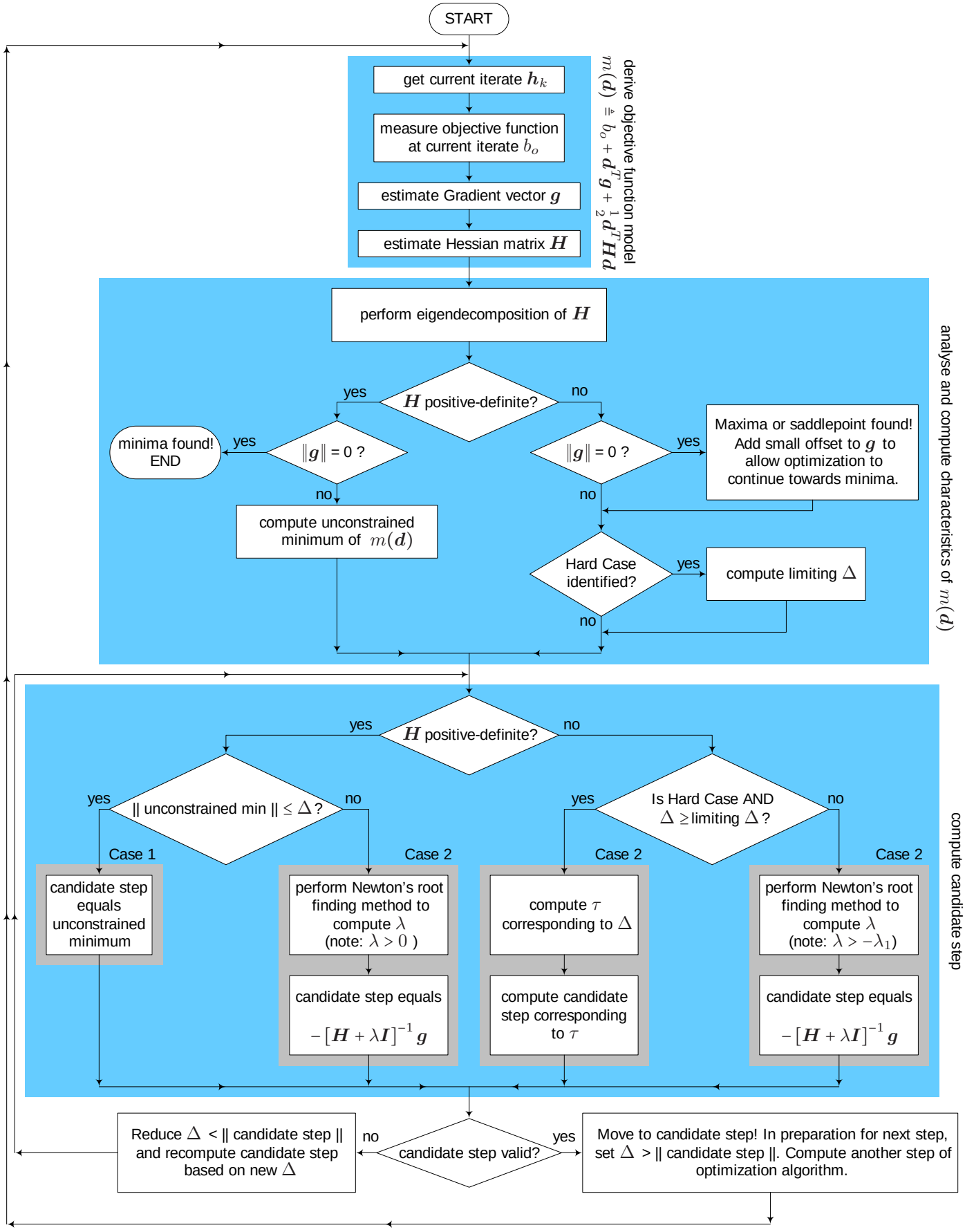


Figure A.2: Flowchart of Trust Region Newton optimization algorithm

A.3 Alpha Branch & Bound

The Alpha Branch & Bound (ABB) optimization algorithm is gradient based with global scope [12]. The algorithm consists of three phases as outlined below.

Phase 1

1. The vector space $\boldsymbol{h}$ is partitioned into boxed regions.
2. Lower & upper bounds on each region's objective function minimum are estimated, a process referred to as *bounding*. It is worth noting that any future reference to region lower and upper bounds is made in this context unless otherwise stated.
3. From the upper bounds of all regions, the overall minimum upper bound is identified. Any region whose lower bound is greater than this overall minimum upper bound is discarded, a process referred to as *fathoming*.

Phase 2

1. From all of the unfathomed regions, the region with the largest bound interval (upper bound subtract lower bound) is identified. This region is bisected along its longest side to give two smaller regions, a process referred to as *branching*.
2. Lower & upper bounds on each new region's objective function minimum are estimated (*bounding*). The two new regions are added to the list of unfathomed regions whilst the original bisected region is discarded.
3. From the upper bounds of all unfathomed regions, the overall minimum upper bound is identified. Any unfathomed region whose lower bound is greater than this overall minimum upper bound is discarded (*fathoming*).
4. Phase 2 is repeated. With smaller regions comes tighter bound intervals and refined region fathoming.

After numerous iterations of Phase 2 (in accordance with Step 4), the number and geometric size of unfathomed regions dramatically reduces. These unfathomed regions subsequently represent candidate regions within which the objective function global minimum could reside. Repetition of Phase 2 ends when the number of unfathomed regions drops to less than some small predefined number κ chosen by the user.

Phase 3

1. For each remaining unfathomed region, a local optimizer is employed to locate the region's objective function minimum. Only one minimum is assumed in each unfathomed region given its small geometric size.

2. Based on all of the minima identified in Step 1, the smallest is considered the objective function global minimum over the entire vector space.

The concepts of Phases 1 – 3 are discussed in further detail in the following. Before continuing however, the reader is encouraged to consult Appendix B.3 to gain familiarity with the eigen properties of the objective function Hessian matrix $\boldsymbol{H}$. Knowledge of these properties will be assumed in the following discussion.

Partitioning The Vector Space Into Boxed Regions

Despite the vector space $\boldsymbol{h} \in \mathbb{R}^n$ being theoretically unconstrained, a practical lower constraint $\boldsymbol{h}^L$ and upper constraint $\boldsymbol{h}^U$ is set in order to establish a finite search space:

$$\boldsymbol{h}^L \leq \boldsymbol{h} \leq \boldsymbol{h}^U \tag{A.47}$$

By dividing each dimension $1 \leq i \leq n$ of the search space into K equal intervals:

$$\begin{matrix} h_i^U - h_i^L \\ K \end{matrix} \tag{A.48}$$

the search space can be partitioned into K^n boxed regions, each with a domain of the form:

$$\left[\begin{array}{ccccc} h_1^L + k_1\left(\begin{matrix} h_1^U - h_1^L \\ K \end{matrix}\right) & \leq & h_1 & \leq & h_1^L + (k_1+1)\left(\begin{matrix} h_1^U - h_1^L \\ K \end{matrix}\right) \\ h_2^L + k_2\left(\begin{matrix} h_2^U - h_2^L \\ K \end{matrix}\right) & \leq & h_2 & \leq & h_2^L + (k_2+1)\left(\begin{matrix} h_2^U - h_2^L \\ K \end{matrix}\right) \\ \vdots & & \vdots & & \vdots \\ h_n^L + k_n\left(\begin{matrix} h_n^U - h_n^L \\ K \end{matrix}\right) & \leq & h_n & \leq & h_n^L + (k_n+1)\left(\begin{matrix} h_n^U - h_n^L \\ K \end{matrix}\right) \end{array}\right] \quad \text{for integers} \quad 0 \leq k_i < K \tag{A.49}$$

Since the domain of each boxed region is dependent on the vector:

$$\boldsymbol{k} = [k_1, k_2, \cdots, k_n] \quad \text{for integers} \quad 0 \leq k_i < K \tag{A.50}$$

each region can be uniquely identified as $R_{\boldsymbol{k}}$ with domain:

$$\left[\boldsymbol{h}^{L,R_{\boldsymbol{k}}} \leq \boldsymbol{h} \leq \boldsymbol{h}^{U,R_{\boldsymbol{k}}}\right] \tag{A.51}$$

where the lower constraint $\boldsymbol{h}^{L,R_{\boldsymbol{k}}}$ and upper constraint $\boldsymbol{h}^{U,R_{\boldsymbol{k}}}$ represent the left and right hand side respectively of (A.49) for the specific $\boldsymbol{k}$ vector:

$$\boldsymbol{h}^{L,R_{\boldsymbol{k}}} = \begin{bmatrix} h_1^L + k_1\begin{pmatrix} h_1^U - h_1^L \\ K \end{pmatrix} \\ h_2^L + k_2\begin{pmatrix} h_2^U - h_2^L \\ K \end{pmatrix} \\ \vdots \\ h_n^L + k_n\begin{pmatrix} h_n^U - h_n^L \\ K \end{pmatrix} \end{bmatrix} \quad \boldsymbol{h}^{U,R_{\boldsymbol{k}}} = \begin{bmatrix} h_1^L + (k_1+1)\begin{pmatrix} h_1^U - h_1^L \\ K \end{pmatrix} \\ h_2^L + (k_2+1)\begin{pmatrix} h_2^U - h_2^L \\ K \end{pmatrix} \\ \vdots \\ h_n^L + (k_n+1)\begin{pmatrix} h_n^U - h_n^L \\ K \end{pmatrix} \end{bmatrix} \tag{A.52}$$

In order to keep the initial number of boxed regions to a practical limit, knowing that each region needs to be bounded in Phase 1 Step 2, K is chosen reasonably small. In this sense, the idea is to partition the search space fairly coarsely and then let the algorithm in Phase 2 determine which regions to further branch and bound.

It is worth noting that, as a consequence of the algorithm's repeated Phase 2 Step 2, these partitioned regions $R_{\boldsymbol{k}}$ will be progressively bisected into smaller boxed regions. In general, these smaller boxed regions do not take the specific form of (A.49) and therefore cannot be uniquely identified by the vector subscript $\boldsymbol{k}$. For this reason, the remaining discussion will be in terms of the general boxed region R_g whose domain is of the form:

$$\left[\boldsymbol{h}^{L,R_g} \leq \boldsymbol{h} \leq \boldsymbol{h}^{U,R_g}\right] = \begin{bmatrix} h_1^{L,R_g} & \leq & h_1 & \leq & h_1^{U,R_g} \\ h_2^{L,R_g} & \leq & h_2 & \leq & h_2^{U,R_g} \\ \vdots & & \vdots & & \vdots \\ h_n^{L,R_g} & \leq & h_n & \leq & h_n^{U,R_g} \end{bmatrix} \tag{A.53}$$

Since the region resides within the finite search space, its lower constraint $\boldsymbol{h}^{L,R_g}$ and upper constraint $\boldsymbol{h}^{U,R_g}$ satisfies:

$$\boldsymbol{h}^L \leq \boldsymbol{h}^{L,R_g} \leq \boldsymbol{h}^{U,R_g} \leq \boldsymbol{h}^U \tag{A.54}$$

Bounding

Given a general boxed region R_g, *bounding* is the process of estimating a lower and upper bound on that region's objective function minimum:

$$\min_{\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}} B(\boldsymbol{h}) \tag{A.55}$$

When estimating a lower bound on the region's objective function minimum, the idea is to engineer a convex function $\mathcal{L}(\boldsymbol{h})$ which under-estimates the objective function $B(\boldsymbol{h})$ over the entire domain of the region. Convexity ensures that $\mathcal{L}(\boldsymbol{h})$ has a single minimum within the region's domain and under-estimation ensures that this minimum is less than the region's objective function minimum, thereby representing a true lower bound. A local optimization algorithm is employed to locate the single minimum of $\mathcal{L}(\boldsymbol{h})$ within the region.

Let $\boldsymbol{\Phi}$ be defined as the diagonal matrix:

$$\boldsymbol{\Phi} = \begin{bmatrix} \alpha_1 & 0 & \cdots & 0 \\ 0 & \alpha_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_n \end{bmatrix} \quad \text{where} \quad \alpha_i \ge 0 \tag{A.56}$$

The convex under-estimating function $\mathcal{L}(\boldsymbol{h})$ then takes the general form [4]:

$$\mathcal{L}(\boldsymbol{h}) = B(\boldsymbol{h}) + \left(\boldsymbol{h}^{L,R_g} - \boldsymbol{h}\right)^T \boldsymbol{\Phi} \left(\boldsymbol{h}^{U,R_g} - \boldsymbol{h}\right) \tag{A.57a}$$

$$= B(\boldsymbol{h}) + \sum_{i=1}^{n} \alpha_i \left(h_i^{L,R_g} - h_i\right)\left(h_i^{U,R_g} - h_i\right) \tag{A.57b}$$

Four points are worth noting about (A.57):

1. With $\alpha_i \ge 0$ and $\left[\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}\right]$, the right hand summation term of (A.57b) is seen to be nonpositive over the entire region, hence ensuring $\mathcal{L}(\boldsymbol{h})$ is a valid under-estimator of $B(\boldsymbol{h})$. $\mathcal{L}(\boldsymbol{h})$ matches $B(\boldsymbol{h})$ at all region corner points.

2. The right hand quadratic term of (A.57a) can be interpreted geometrically as the convex quadratic surface $\theta(\boldsymbol{h}) = \boldsymbol{h}^T \boldsymbol{\Phi}\, \boldsymbol{h}$ translated to the region's centroid $\left(\boldsymbol{h}^{U,R_g} + \boldsymbol{h}^{L,R_g}\right)/2$ and then de-elevated to ensure negativity over the entire

region. That is:

$$\left(\boldsymbol{h}^{L,R_g} - \boldsymbol{h}\right)^T \boldsymbol{\Phi} \left(\boldsymbol{h}^{U,R_g} - \boldsymbol{h}\right) = \theta\left(\boldsymbol{h} - \begin{bmatrix} \boldsymbol{h}^{U,R_g} + \boldsymbol{h}^{L,R_g} \\ 2 \end{bmatrix}\right) - \theta\left(\boldsymbol{h}^{L,R_g} - \begin{bmatrix} \boldsymbol{h}^{U,R_g} + \boldsymbol{h}^{L,R_g} \\ 2 \end{bmatrix}\right) \quad \text{(A.58)}$$

In this sense, the Hessian matrix of the right hand quadratic term of (A.57a) is simply the Hessian matrix of $\theta(\boldsymbol{h}) = \boldsymbol{h}^T\boldsymbol{\Phi}\,\boldsymbol{h} = \left(\frac{1}{2}\right)\boldsymbol{h}^T(2\boldsymbol{\Phi})\,\boldsymbol{h}$ which is $2\boldsymbol{\Phi}$. It logically follows from (A.57a) and the linearity property of derivatives that the Hessian matrix of $\mathcal{L}(\boldsymbol{h})$ is given by:

$$\boldsymbol{H}_{\mathcal{L}}(\boldsymbol{h}) = \boldsymbol{H}_B(\boldsymbol{h}) + 2\boldsymbol{\Phi} \quad \text{(A.59)}$$

where $\boldsymbol{H}_B(\boldsymbol{h})$ represents the Hessian matrix of the objective function $B(\boldsymbol{h})$.

3. The values of α_i are chosen specifically to ensure regional convexity of $\mathcal{L}(\boldsymbol{h})$. To achieve regional convexity, $\boldsymbol{H}_{\mathcal{L}}(\boldsymbol{h})$ must be positive-definite and therefore possess all positive eigenvalues (Property 4, Appendix B.3) for $\left[\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}\right]$.

   From (A.59) and Gerschgorin's Theorem (Property 8, Appendix B.3), $\boldsymbol{H}_{\mathcal{L}}(\boldsymbol{h})$'s i^{th} eigenvalue $\lambda_i(\boldsymbol{h})$ is bounded by:

$$\left(H_{B\,i,i}(\boldsymbol{h}) - \sum_{\substack{j=1 \\ j\neq i}}^{n}\left|H_{B\,i,j}(\boldsymbol{h})\right|\right) + 2\alpha_i \;\le\; \lambda_i(\boldsymbol{h}) \;\le\; \left(H_{B\,i,i}(\boldsymbol{h}) + \sum_{\substack{j=1 \\ j\neq i}}^{n}\left|H_{B\,i,j}(\boldsymbol{h})\right|\right) + 2\alpha_i \quad \text{(A.60)}$$

   where $H_{B\,i,j}(\boldsymbol{h})$ is the $(i,j)^{\text{th}}$ element of $\boldsymbol{H}_B(\boldsymbol{h})$. It is noted that in (A.60), the subscript i of $\lambda_i(\boldsymbol{h})$ signifies the Gerschgorin disk D_i with which the eigenvalue is associated. This is not to be confused with the eigenvalue subscript used in Properties 1 and 2 of Appendix B.3 which denotes the relative magnitude of eigenvalues.

   To ensure the lower bound of (A.60) is positive for all $\left[\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}\right]$, thereby ensure positive eigenvalues and hence regional convexity of $\mathcal{L}(\boldsymbol{h})$, $\alpha_i \ge 0$ must satisfy:

$$\alpha_i \ge \max\left\{0,\, -\frac{1}{2}\left(\min_{\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}}\left(H_{B\,i,i}(\boldsymbol{h}) - \sum_{\substack{j=1 \\ j\neq i}}^{n}\left|H_{B\,i,j}(\boldsymbol{h})\right|\right)\right)\right\} \quad \text{(A.61)}$$

   The fact that α_i is bounded below according to (A.61) can be understood intuitively by recognizing that the α_i's determine the convexity of the right

hand quadratic term of (A.57a). By choosing the α_i's sufficiently large, the convexity of this quadratic term can be made to overpower all nonconvexities in the objective function $B(\boldsymbol{h})$ and hence guarantee convexity of $\mathcal{L}(\boldsymbol{h})$.

4. It is shown in [10,175] that $\mathcal{L}(\boldsymbol{h})$ under-estimates the objective function $B(\boldsymbol{h})$ by a maximum:

$$\max_{\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}} \Big(B(\boldsymbol{h}) - \mathcal{L}(\boldsymbol{h}) \Big) = \frac{1}{4} \sum_{i=1}^{n} \alpha_i \left(h_i^{U,R_g} - h_i^{L,R_g} \right)^2 \tag{A.62}$$

It follows that minimal α_i values and smaller boxed regions lead to tighter under-estimating functions. This in general leads to tighter bound intervals (upper bound subtract lower bound) and hence refined region fathoming.

Based on Dot-Points 3 and 4 above, the ideal value of α_i, that which leads to the tightest convex under-estimating function, is given by:

$$\alpha_i^{\text{ideal}} = \max \left\{ 0, -\frac{1}{2} \left(\min_{\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}} \left(H_{B\,i,i}(\boldsymbol{h}) - \sum_{\substack{j=1 \\ j \ne i}}^{n} \big| H_{B\,i,j}(\boldsymbol{h}) \big| \right) \right) \right\} \tag{A.63}$$

Unfortunately, unless $\boldsymbol{H}_B(\boldsymbol{h})$ is strictly analytic, the right hand minimization term of (A.63) is virtually impossible to compute, leaving α_i^{ideal} unobtainable in practice. In such situations, this problematic right hand minimization term is completely replaced with its practically computable lower bound to give, by definition, a generally nonideal though valid value of α_i, subsequently referred to as α_i^{prac} to denote its practical computability. The lower bound on the right hand minimization term is computed along with α_i^{prac} as follows:

- The region's objective function interval Hessian matrix $[\boldsymbol{H}_B]^{R_g} = \big[\, \underline{\boldsymbol{H}}_B, \bar{\boldsymbol{H}}_B \,\big]$ (Property 7, Appendix B.3) is estimated from numerous samples of $\boldsymbol{H}_B(\boldsymbol{h})$ taken within the region $\big[\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}\big]$. These Hessian samples are themselves estimated via the techniques of Appendix B.2. During the interval estimation process, elements of the $\boldsymbol{H}_B(\boldsymbol{h})$ samples are treated independently [197, 224]. A lower bound on the i^{th} eigenvalue of $\Big\{ [\boldsymbol{H}_B]^{R_g} \Big\}$ is then computed via the principles of Gerschgorin's Theorem as:

$$\underline{H}_{B\,i,i} - \sum_{\substack{j=1 \\ j \ne i}}^{n} \max \Big(\big| \underline{H}_{B\,i,j} \big|, \big| \bar{H}_{B\,i,j} \big| \Big) \tag{A.64}$$

Here $\underline{H}_{B\,i,j}$ and $\bar{H}_{B\,i,j}$ represent the $(i,j)^{\text{th}}$ elements of $\underline{\boldsymbol{H}}_B$ and $\bar{\boldsymbol{H}}_B$ respectively.

- With Gerschgorin's Theorem still in mind, the right hand minimization term of Equation (A.63) is recognized as the lower bound on the i^{th} eigenvalue of $\left\{ \boldsymbol{H}_B(\boldsymbol{h}) \ \forall \ \left[\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g} \right] \right\}$. Since by definition:

$$\left\{ \boldsymbol{H}_B(\boldsymbol{h}) \ \forall \ \left[\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g} \right] \right\} \ \subseteq \ \left\{ \left[\boldsymbol{H}_B \right]^{R_g} \right\} \tag{A.65}$$

it follows that:

$$\underline{H}_{B\,i,i} - \sum_{\substack{j=1 \\ j \neq i}}^{n} \max\left(\left| \underline{H}_{B\,i,j} \right|, \left| \bar{H}_{B\,i,j} \right| \right) \leq \min_{\boldsymbol{h}^{L,R_g} \le \boldsymbol{h} \le \boldsymbol{h}^{U,R_g}} \left(H_{B\,i,i}(\boldsymbol{h}) - \sum_{\substack{j=1 \\ j \neq i}}^{n} \left| H_{B\,i,j}(\boldsymbol{h}) \right| \right) \tag{A.66}$$

and (A.64) represents the sought after lower bound on the right hand minimization term of (A.63).

- The right hand minimization term of (A.63) is then completely replaced with this (A.64) lower bound to give the practically computable α_i^{prac}:

$$\alpha_i^{\text{prac}} = \max\left\{ 0, -\frac{1}{2} \left(\underline{H}_{B\,i,i} - \sum_{\substack{j=1 \\ j \neq i}}^{n} \max\left(\left| \underline{H}_{B\,i,j} \right|, \left| \bar{H}_{B\,i,j} \right| \right) \right) \right\} \tag{A.67}$$

By definition, $\alpha_i^{\text{prac}} \ge \alpha_i^{\text{ideal}}$ and hence convexity condition (A.61) is satisfied.

With α_i computed via (A.67), the general form of $\mathcal{L}(\boldsymbol{h})$ originally expressed in (A.57) is now fully defined. As discussed previously, since $\mathcal{L}(\boldsymbol{h})$ is a convex under-estimator of the objective function $B(\boldsymbol{h})$ over the entire region R_g, its single regional minimum represents the sought after lower bound on the region's objective function minimum. $\mathcal{L}(\boldsymbol{h})$'s single regional minimum is located via a local optimization algorithm operating on (A.57).

A valid upper bound on the region's objective function minimum is estimated simply as the value of the objective function $B(\boldsymbol{h})$ at $\mathcal{L}(\boldsymbol{h})$'s single regional minimum.

Fathoming

Given a set of general boxed regions and the corresponding lower and upper bounds on each region's objective function minimum, *fathoming* is the process of identifying and discarding regions which cannot theoretically contain the objective function's global minimum. The identification process is based on comparison of each region's lower bound to the overall minimum upper bound.

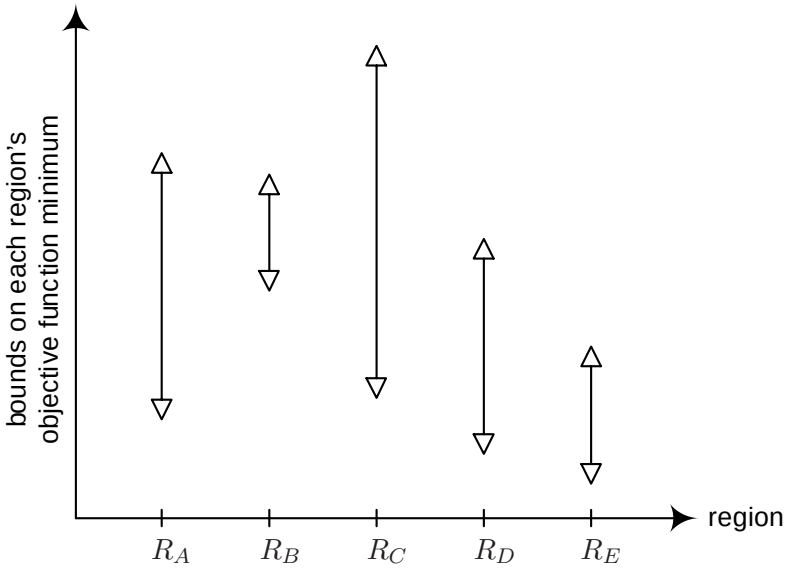


Figure A.3: Fathoming example for the Alpha Branch & Bound algorithm

For example, consider the set of general boxed regions $\{ R_A, R_B, R_C, R_D, R_E \}$ and let the lower and upper bounds on each region's objective function minimum be represented graphically in Figure A.3. Here, $\triangledown$ and $\triangle$ represent the lower and upper bounds respectively for each region whilst the continuous line connecting $\triangledown$ and $\triangle$ represents the corresponding bound interval. From the upper bounds of all regions in Figure A.3, the overall minimum upper bound is seen to belong to R_E. Since the lower bound of R_B is greater than the upper bound of R_E, it logically follows that R_B's objective function minimum must be theoretically greater than R_E's objective function minimum and therefore R_B cannot possibly contain the global objective function minimum. In this case, R_B can be discarded from further analysis. R_B is then said to be fathomed.

At this point in the example, nothing more can be ascertained about the location of the global objective function minimum from the remaining unfathomed set of region's and bounds and therefore *branching* must take place.

Branching

Given a set of unfathomed general boxed regions and the corresponding lower and upper bounds on each region's objective function minimum, *branching* is the process of identifying the region with the largest bound interval (upper bound subtract lower bound) and bisecting this region along its longest side to give two smaller regions.

Continuing on from the previous *fathoming* example, the following four unfathomed regions remain $\{ R_A, R_C, R_D, R_E \}$. The lower and upper bounds on each region's objective function minimum are taken from Figure A.3 and repeated in

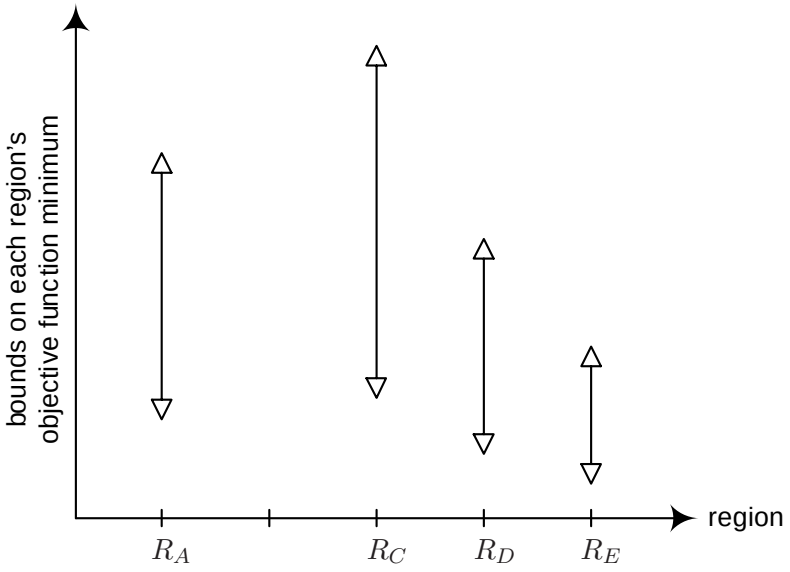


Figure A.4: Branching example for the Alpha Branch & Bound algorithm

Figure A.4. Once again, $\triangledown$ and $\triangle$ represent the lower and upper bounds respectively for each region whilst the continuous line connecting $\triangledown$ and $\triangle$ represents the corresponding bound interval. From the bound intervals of all regions in Figure A.4, the largest bound interval is seen to belong to R_C. Let the domain of region R_C be represented by:

$$\left[\boldsymbol{h}^{L,R_C} \leq \boldsymbol{h} \leq \boldsymbol{h}^{U,R_C}\right] = \begin{bmatrix} h_1^{L,R_C} & \leq & h_1 & \leq & h_1^{U,R_C} \\ \vdots & & \vdots & & \vdots \\ h_l^{L,R_C} & \leq & h_l & \leq & h_l^{U,R_C} \\ \vdots & & \vdots & & \vdots \\ h_n^{L,R_C} & \leq & h_n & \leq & h_n^{U,R_C} \end{bmatrix} \tag{A.68}$$

where $h_l^{U,R_C} - h_l^{L,R_C}$ represents the region's longest side, that is:

$$l = \arg\max_i \left(h_i^{U,R_C} - h_i^{L,R_C}\right) \tag{A.69}$$

Region R_C is then bisected along its l^{th} side to give two new regions:

$$\left[\begin{array}{lcl} h_1^{L,R_C} \le h_1 \le h_1^{U,R_C} \\ \vdots \quad \vdots \quad \vdots \\ h_l^{L,R_C} \le h_l \le \dfrac{\left(h_l^{U,R_C}+h_l^{L,R_C}\right)}{2} \\ \vdots \quad \vdots \quad \vdots \\ h_n^{L,R_C} \le h_n \le h_n^{U,R_C} \end{array}\right] \left[\begin{array}{lcl} h_1^{L,R_C} \le h_1 \le h_1^{U,R_C} \\ \vdots \quad \vdots \quad \vdots \\ \dfrac{\left(h_l^{U,R_C}+h_l^{L,R_C}\right)}{2} \le h_l \le h_l^{U,R_C} \\ \vdots \quad \vdots \quad \vdots \\ h_n^{L,R_C} \le h_n \le h_n^{U,R_C} \end{array}\right] \quad \text{(A.70)}$$

This Alpha Branch & Bound algorithm is summarized in the flowchart of Figure A.5. It is worth noting that the additional Phase 1 yellow steps are optional. If the user has application-specific *a priori* knowledge that outer $R_{\boldsymbol{k}}$ corner regions (those with a large centroid norm) don't possess the objective function global minimum, then such regions can be discarded from the very outset, thereby reducing the algorithm's initial bounding load and hence increasing initial algorithm speed.

This algorithm is implemented on the laboratory transmitter testbed via the software template function *AlphaBranchBound()*. Corresponding declaration and definition source code resides in project files *AlphaBranchBoundOptimization.h* and *AlphaBranchBoundOptimization Templates.cpp* respectively. Both files are located within folder *Software\Cpp* on the accompanying DVD.

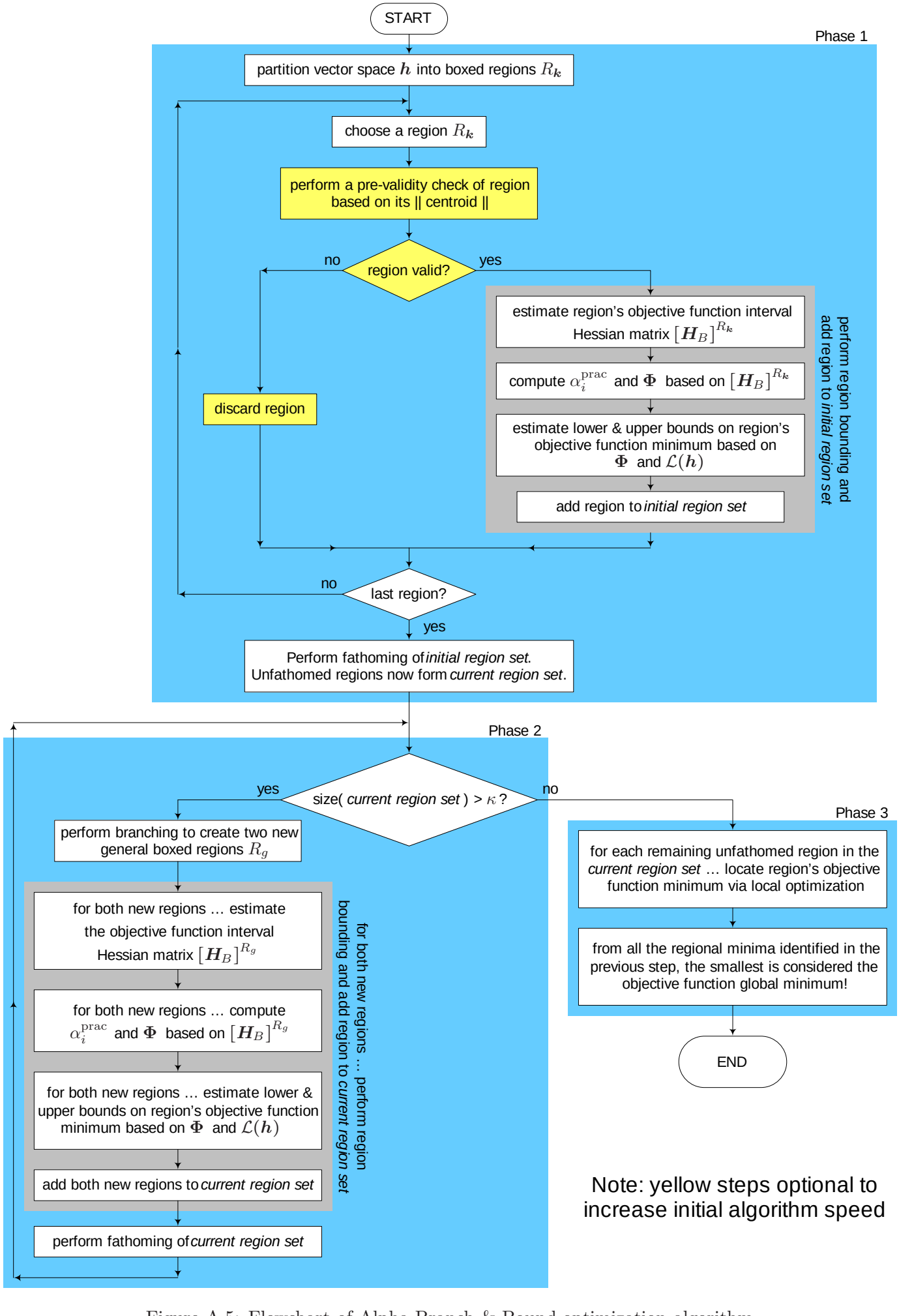


Figure A.5: Flowchart of Alpha Branch & Bound optimization algorithm

A.4 Nelder-Mead Simplex

Developed by J. A. Nelder and R. Mead [196] from the earlier work of [260], this optimization algorithm uses basic simplex geometry plus objective function measurement and comparison to identify a local objective function minimum. A simplex is simply the convex hull of a set of $(n+1)$ points in the $\mathbb{R}^n$ vector space [111]. A triangle for example is a simplex in the $\mathbb{R}^2$ space, likewise a tetrahedron is a simplex in the $\mathbb{R}^3$ space. In its most basic form, the optimization algorithm involves:

- initially defining an $(n+1)$ vertex simplex in the $\mathbb{R}^n$ vector space about the starting point of the optimization.
- updating the geometry of this simplex at each optimization step, via either *reflection*, *contraction*, *expansion* or *shrinkage*, with the general goal of reducing its vertex objective function values. Viewed abstractly in terms of the local objective function landscape, this geometric updating process sees the simplex elongate down long inclined planes, change direction on encountering a valley at an angle and contract in the neighborhood of a minimum [196].

In the context of the first Dot-Point above, let the initial simplex S_0 be defined by the set of $(n+1)$ vertices:

$$S_0 = \left\{ \boldsymbol{h}^s, \quad \boldsymbol{h}^s + \begin{bmatrix} d \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \quad \boldsymbol{h}^s + \begin{bmatrix} 0 \\ d \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \quad \cdots \quad , \quad \boldsymbol{h}^s + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ d \\ 0 \end{bmatrix}, \quad \boldsymbol{h}^s + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ d \end{bmatrix} \right\} \tag{A.71}$$

where $\boldsymbol{h}^s$ is the vector space starting point of the mathematical optimization and d is a small positive increment.

In the context of the second Dot-Point above, let S_k represent the geometrically updated simplex at optimization step k $(k \geq 1)$. S_k is then computed as follows:

1. Vertices of the previous simplex S_{k-1} are uniquely labeled according to their relative objective function values. The vertex exhibiting the lowest objective function value is labeled $\boldsymbol{v}_1$, the vertex exhibiting the second lowest objective function value is labeled $\boldsymbol{v}_2$ and so on up to the vertex exhibiting the highest objective function value which is labeled $\boldsymbol{v}_{n+1}$. That is:

$$B(\boldsymbol{v}_1) \leq B(\boldsymbol{v}_2) \leq \cdots \leq B(\boldsymbol{v}_n) \leq B(\boldsymbol{v}_{n+1}) \tag{A.72}$$

2. The centroid of S_{k-1}'s best n vertices is computed as:

$$\bar{\boldsymbol{v}} = \binom{1}{n} \sum_{i=1}^{n} \boldsymbol{v}_i \tag{A.73}$$

3. The *reflection point* of S_{k-1} is defined and computed as:

$$\boldsymbol{v}_r = \bar{\boldsymbol{v}} - (\boldsymbol{v}_{n+1} - \bar{\boldsymbol{v}}) \tag{A.74}$$

   This *reflection point* lies along the line intersecting the centroid $\bar{\boldsymbol{v}}$ and the worst vertex $\boldsymbol{v}_{n+1}$. $B(\boldsymbol{v}_r)$ is then measured.

4. The desired S_k is formed by either *reflecting*, *contracting*, *expanding* or *shrinking* S_{k-1} based on the value of $B(\boldsymbol{v}_r)$ relative to $B(\boldsymbol{v}_1)$, $B(\boldsymbol{v}_n)$ and $B(\boldsymbol{v}_{n+1})$:

   If $B(\boldsymbol{v}_1) \leq B(\boldsymbol{v}_r) < B(\boldsymbol{v}_n)$

   S_k is formed by *reflecting* S_{k-1}. That is, S_{k-1}'s worst vertex $\boldsymbol{v}_{n+1}$ is replaced with the *reflection point* $\boldsymbol{v}_r$.

   If $B(\boldsymbol{v}_r) < B(\boldsymbol{v}_1)$

   The objective function is measured at the *expansion point* defined as:

$$\boldsymbol{v}_e = \bar{\boldsymbol{v}} - 2\,(\boldsymbol{v}_{n+1} - \bar{\boldsymbol{v}}) \tag{A.75}$$

   If $B(\boldsymbol{v}_e) < B(\boldsymbol{v}_r)$

   S_k is formed by *expanding* S_{k-1}. That is, S_{k-1}'s worst vertex $\boldsymbol{v}_{n+1}$ is replaced with the *expansion point* $\boldsymbol{v}_e$.

   Otherwise

   S_k is once again formed by *reflecting* S_{k-1}. That is, S_{k-1}'s worst vertex $\boldsymbol{v}_{n+1}$ is replaced with the *reflection point* $\boldsymbol{v}_r$.

   If $B(\boldsymbol{v}_n) \leq B(\boldsymbol{v}_r) < B(\boldsymbol{v}_{n+1})$

   The objective function is measured at the *outside contraction point* defined as:

$$\boldsymbol{v}_{oc} = \bar{\boldsymbol{v}} - \frac{1}{2}\,(\boldsymbol{v}_{n+1} - \bar{\boldsymbol{v}}) \tag{A.76}$$

   If $B(\boldsymbol{v}_{oc}) \leq B(\boldsymbol{v}_r)$

   S_k is formed by *outside contracting* S_{k-1}. That is, S_{k-1}'s worst vertex $\boldsymbol{v}_{n+1}$ is replaced with the *outside contraction point* $\boldsymbol{v}_{oc}$.

   Otherwise

   S_k is formed by *shrinking* S_{k-1}. That is, S_{k-1}'s best vertex $\boldsymbol{v}_1$ is kept while all remaining vertices $\boldsymbol{v}_i$ $[2 \leq i \leq (n+1)]$ are replaced with $\frac{1}{2}(\boldsymbol{v}_1 + \boldsymbol{v}_i)$.

If $B(\boldsymbol{v}_r) \geq B(\boldsymbol{v}_{n+1})$

The objective function is measured at the *inside contraction point* defined as:

$$\boldsymbol{v}_{ic} = \bar{\boldsymbol{v}} + \frac{1}{2}(\boldsymbol{v}_{n+1} - \bar{\boldsymbol{v}}) \tag{A.77}$$

If $B(\boldsymbol{v}_{ic}) < B(\boldsymbol{v}_{n+1})$

S_k is formed by *inside contracting* S_{k-1}. That is, S_{k-1}'s worst vertex $\boldsymbol{v}_{n+1}$ is replaced with the *inside contraction point* $\boldsymbol{v}_{ic}$.

Otherwise

S_k is once again formed by *shrinking* S_{k-1}. That is, S_{k-1}'s best vertex $\boldsymbol{v}_1$ is kept while all remaining vertices $\boldsymbol{v}_i$ $[2 \leq i \leq (n+1)]$ are replaced with $\frac{1}{2}(\boldsymbol{v}_1 + \boldsymbol{v}_i)$.

Several points are worth noting about this optimization algorithm:

- Just like the *reflection point* (A.74), all of the *expansion* (A.75), *outside contraction* (A.76) and *inside contraction points* (A.77) lie along the line intersecting $\bar{\boldsymbol{v}}$ and $\boldsymbol{v}_{n+1}$ of S_{k-1}. This makes the updating process, and hence the entire algorithm, geometrically deterministic.

- Geometric updating of the simplex at each optimization step is a *Gradient-Free* process, involving only objective function measurement and comparison. For this reason, the algorithm is considered highly efficient compared to the *Gradient based* optimizers discussed previously. The number of objective function measurements required by each form of geometric update is outlined below:

Geometric Update of Simplex	**Objective Function Measurements**
Reflection	maximum of 2
Expansion	2
Outside Contraction	2
Inside Contraction	2
Shrinkage	n+2

Table A.1: Objective function measurements per geometric update

- Whilst technically a local optimization algorithm, a more global emphasis can be achieved if the initial simplex S_0 is defined geometrically large by increasing d of (A.71). This gives the simplex a chance to explore more of the vector space before *contracting* and *shrinking*.

- After long periods of simplex updating, a situation known as *simplex degeneration* or *stagnation* may occur whereby vertices of the simplex become ill aligned (determinant of simplex edge matrix approaches zero) and optimization convergence prematurely ceases [133, 200]. When this occurs, there is no choice but to reset the simplex according to (A.71). Practical implementations of the algorithm take a proactive approach to this problem. They restart the algorithm on a regular basis and hence deny the simplex any opportunity of becoming degenerate.

- Convergence theory of this seemingly intuitive algorithm is given in [134, 148].

At optimization step k, automated exiting from the algorithm is based on the following mean square criterion [196]:

$$\Upsilon = \binom{1}{n+1} \sum_{i=1}^{n+1} \Big[B(S_k \,\text{centroid}) - B(S_k \,\text{vertex}\ i) \Big]^2 \tag{A.78}$$

A large value of Υ generally indicates the simplex is situated on an objective function slope whilst a small value of Υ generally indicates that the simplex has flattened out (and contracted) into an objective function minimum region. Based on this reasoning, if Υ drops below some small predefined value Υ_0, the algorithm exits and returns S_k's best vertex.

This Nelder-Mead Simplex algorithm is summarized in the flowchart of Figure A.6. It is also implemented on the laboratory transmitter testbed via the software template function *NelderMeadSimplex()*. Corresponding declaration and definition source code resides in project files *NelderMeadSimplexOptimization.h* and *NelderMeadSimplexOptimization Templates.cpp* respectively. Both files are located within folder *Software\Cpp* on the accompanying DVD.

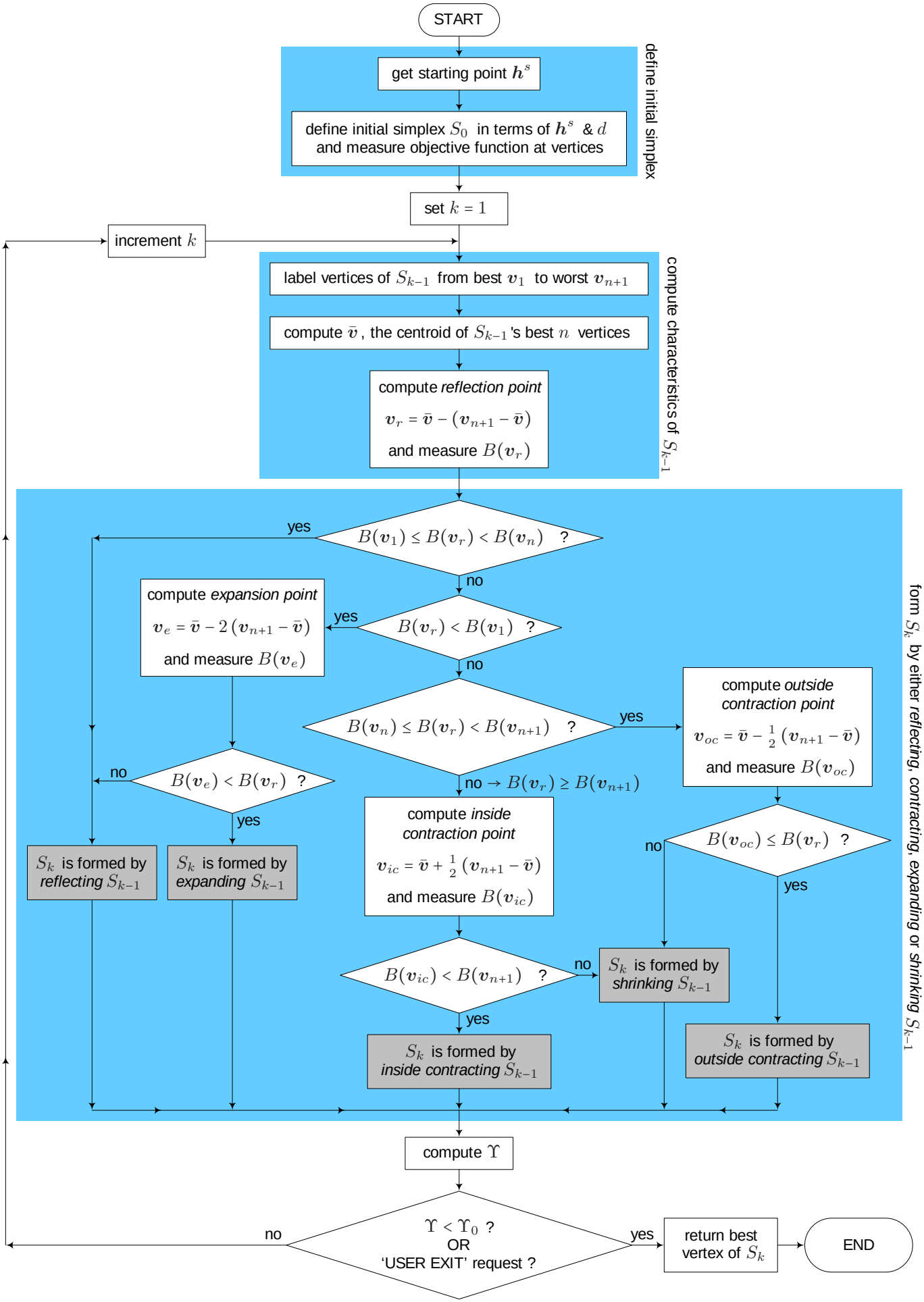


Figure A.6: Flowchart of Nelder-Mead Simplex optimization algorithm

A.5 Genetic

Rooted in the biological sciences [20, 21, 76, 109], this optimization algorithm abstracts the general principles of evolution (*natural selection*, *reproduction* and *survival of the fittest*) to generate an estimate of the objective function global minimum [16,17,87]. In keeping with the literature, it is important to note the following changes in terminology as a result of the algorithm's biological roots [283, 284].

A point in the optimization vector space is now called a *chromosome* rather than a variable vector, a single element of a variable vector is now called a *gene* rather than a variable and the function to be minimized is now called a *fitness function* rather than an objective function. In this sense, a *fitter* chromosome has a smaller objective function value.

Figure A.7 outlines the Genetic algorithm in its most basic form. Central to the algorithm's operation, the *Population Pool* represents a set of ξ candidate solutions to the global minimization problem. After its initial estimation, the *Population Pool* is iteratively refined via a biological evolutionary process (involving *selection*, *reproduction*, *repopulation*) until chromosome concentrations form. These concentrations, representing regions of objective function minima, are subsequently used to estimate the desired global minimum. Such overview concepts are discussed in more detail in the following.

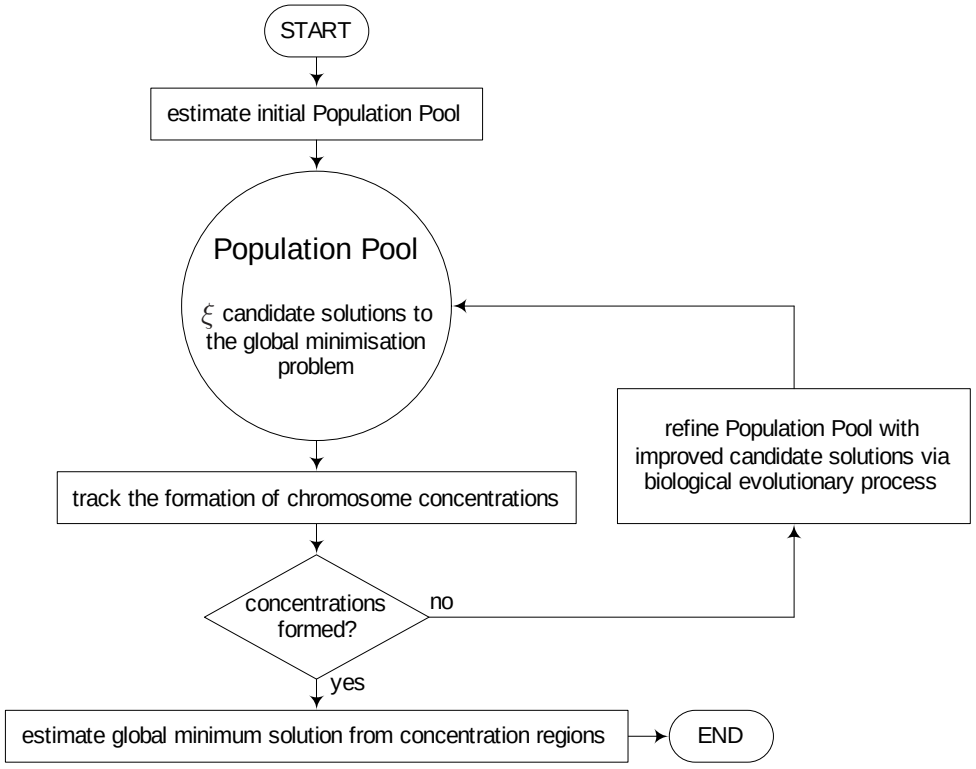


Figure A.7: Overview of Genetic optimization algorithm

Estimating the Initial Population Pool

When estimating the initial Population Pool, the goal is not to *pinpoint* the global minimum but rather to assemble a set of ξ *fit* chromosomes from across the entire vector space. This initial Population Pool merely seeds the algorithm.

In this sense, the simplest method of estimating the initial Population Pool is to randomly generate $\chi \gg \xi$ chromosomes and measure their fitness. From these χ chromosomes, the ξ fittest form the Population Pool. An alternative more structured method is to compute a uniform grid of $\chi \gg \xi$ chromosomes across the vector space and measure their fitness. Once again, the ξ fittest form the Population Pool.

Refining the Population Pool Via Biological Evolution

The biological evolutionary process used to refine the Population Pool consists of three steps as outlined in Figure A.8:

Step 1 - Selection The κ fittest chromosomes of the Population Pool are selected to form the *Mating Pool*. Chromosomes of the *Mating Pool* are eligible for reproduction. This step abstracts the *natural selection* principle of biological evolution.

Step 2 - Reproduction ϱ randomly selected pairs of Mating Pool chromosomes are chosen to mate, with each mating pair reproducing a single chromosome. A reproduced chromosome is referred to as an *offspring* whilst each chromosome of the mating pair is called a *parent*. The x^{th} gene of an offspring chromosome is created by:

1. randomly selecting one of the two parent chromosomes and inheriting its x^{th} gene (a process referred to as *crossover*).
2. introducing a small random change to this inherited gene (a process referred to as *mutation*). In general, the small random change is modeled with a Normal Distribution.

The entire set of ϱ offspring generated from this reproduction step form the *Offspring Pool*. As its name suggests, this step abstracts the *reproduction* principle of biological evolution.

Step 3 - Repopulation All ϱ chromosomes of the Offspring Pool are added to the current Population Pool. This appended Population Pool is then truncated back to the ξ fittest chromosomes. This step abstracts the *survival of the fittest* principle of biological evolution.

Population pool refinement is attributed to the fact that only the fittest chromosomes are mated during the evolutionary process.

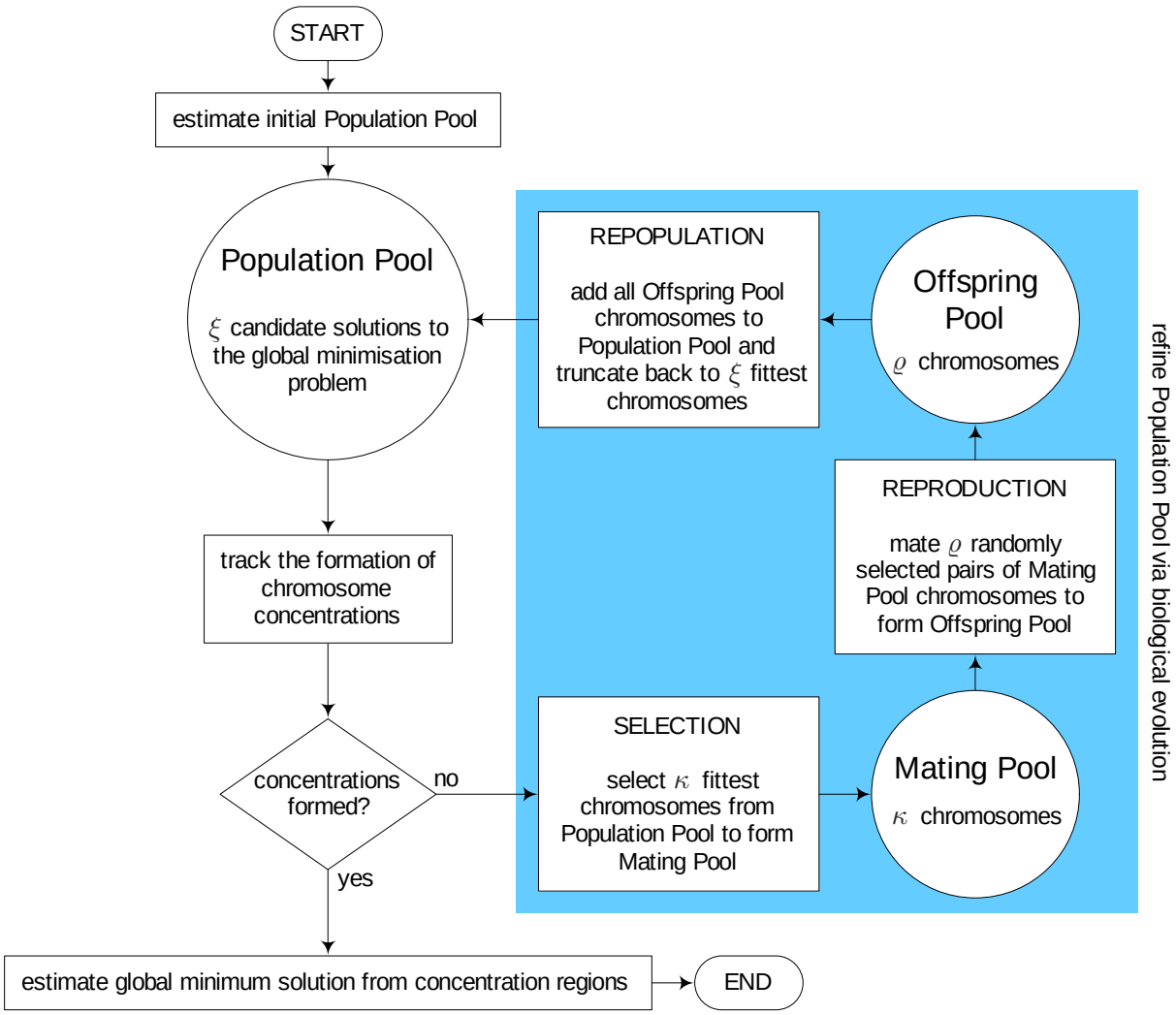


Figure A.8: Biological evolutionary process used to refine the Population Pool

Tracking The Formation of Chromosome Concentrations

Repeated evolutionary refinement of the Population Pool causes chromosomes to become fitter and hence gradually concentrate in regions of objective function minima. Mathematically speaking, a chromosome concentration is defined as a set of chromosomes exhibiting similar individual gene values. By searching for similarities in gene values, pattern recognition algorithms can subsequently identify and track the formation of individual chromosome concentrations.

The chromosome concentration formation process can be demonstrated visually, as follows, via indicative gene plots of the Population Pool after short, medium and long-term refinement.

After short-term refinement (Figure A.9a), the Population Pool's chromosomes are still fairly raw and scattered throughout the entire vector space. As a result, a plot of the Population Pool appears random with no sign of chromosome concentrations.

After medium-term refinement (Figure A.9b), the Population Pool's chromosomes are substantially refined with distinct chromosome concentrations forming. One concentration represents the global minimum region whilst the others represent local minima regions. In the example of Figure A.9b, three concentrations exist.

After long-term refinement (Figure A.9c), the Population Pool is refined to the point where all chromosomes concentrate around the global minimum. Evolutionary refinement has effectively run to completion. Those previous medium-term concentrations representing local minima regions have gradually dispersed (in order of least fit) with their chromosomes reconcentrating around this global minimum.

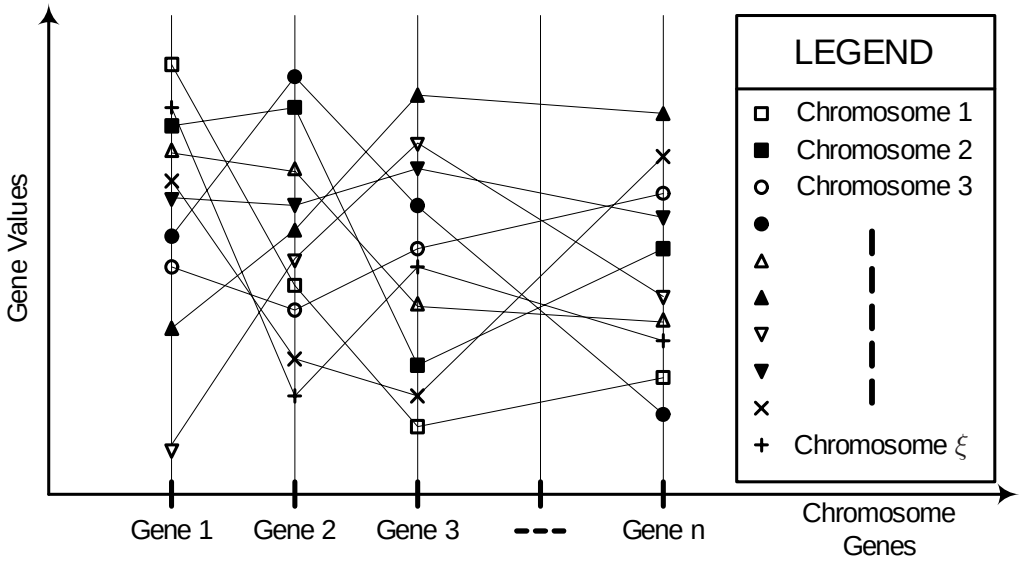
Ideally, with the above in mind, the goal would be to run evolutionary refinement to completion (long-term refinement) such that a single global minimum region is identified. This is impractical however given finite time constraints. Practical implementations of the algorithm therefore halt refinement once a small number ϑ of distinct chromosome concentrations have formed (medium-term refinement) and estimate the desired global minimum from these ϑ concentrations via regional local optimization and comparison as discussed in the next section.

Estimating the Global Minimum From the Concentration Regions

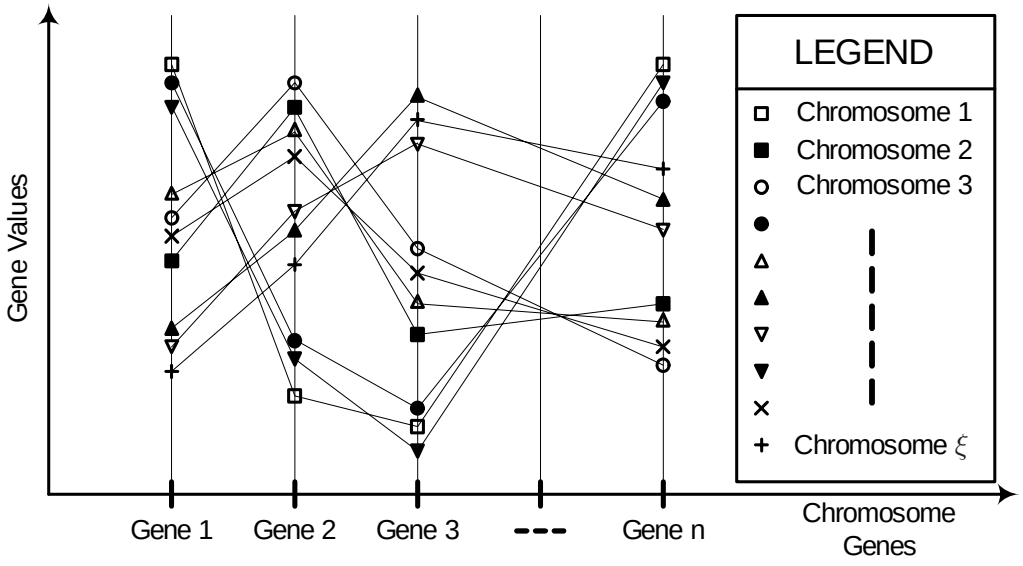
Estimating the desired global minimum from the small number ϑ of distinct chromosome concentrations is a three step process:

1. The centroid of each chromosome concentration is computed.
2. A local optimizer is employed at each computed centroid to locate the corresponding local minimum.
3. The desired global minimum is estimated as the smallest of these ϑ local minima.

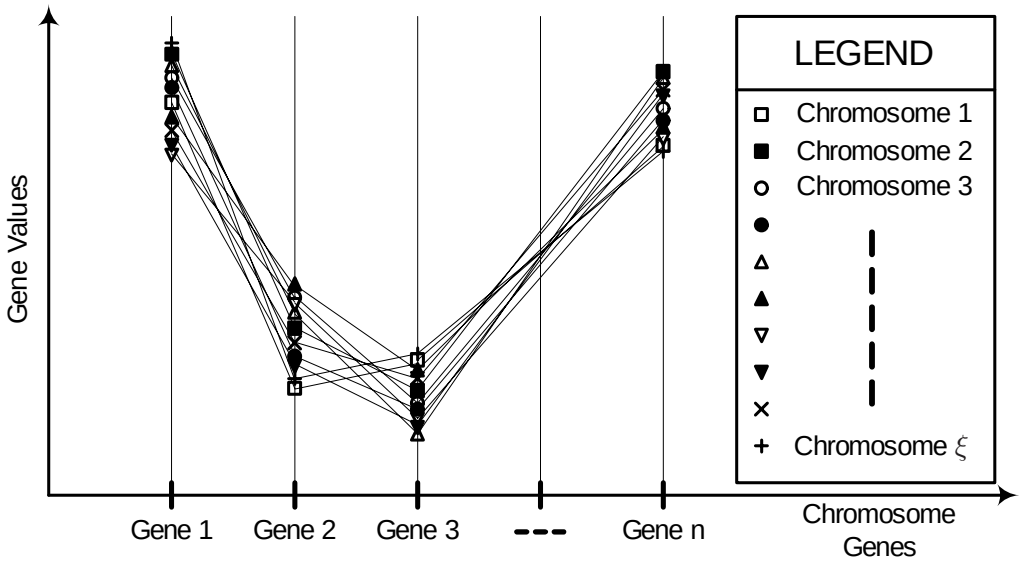
This Genetic algorithm is summarized in the flowchart of Figure A.10. It is also implemented on the laboratory transmitter testbed via the template function *Genetic()*. Corresponding declaration and definition source code resides in the project files *GeneticOptimization.h* and *GeneticOptimization Templates.cpp* respectively. Both files are located within folder *Software\Cpp* on the accompanying DVD.



(a)



(b)



(c)

Figure A.9: Gene plots of the Population Pool after (a) short-term (b) medium-term and (c) long-term refinement

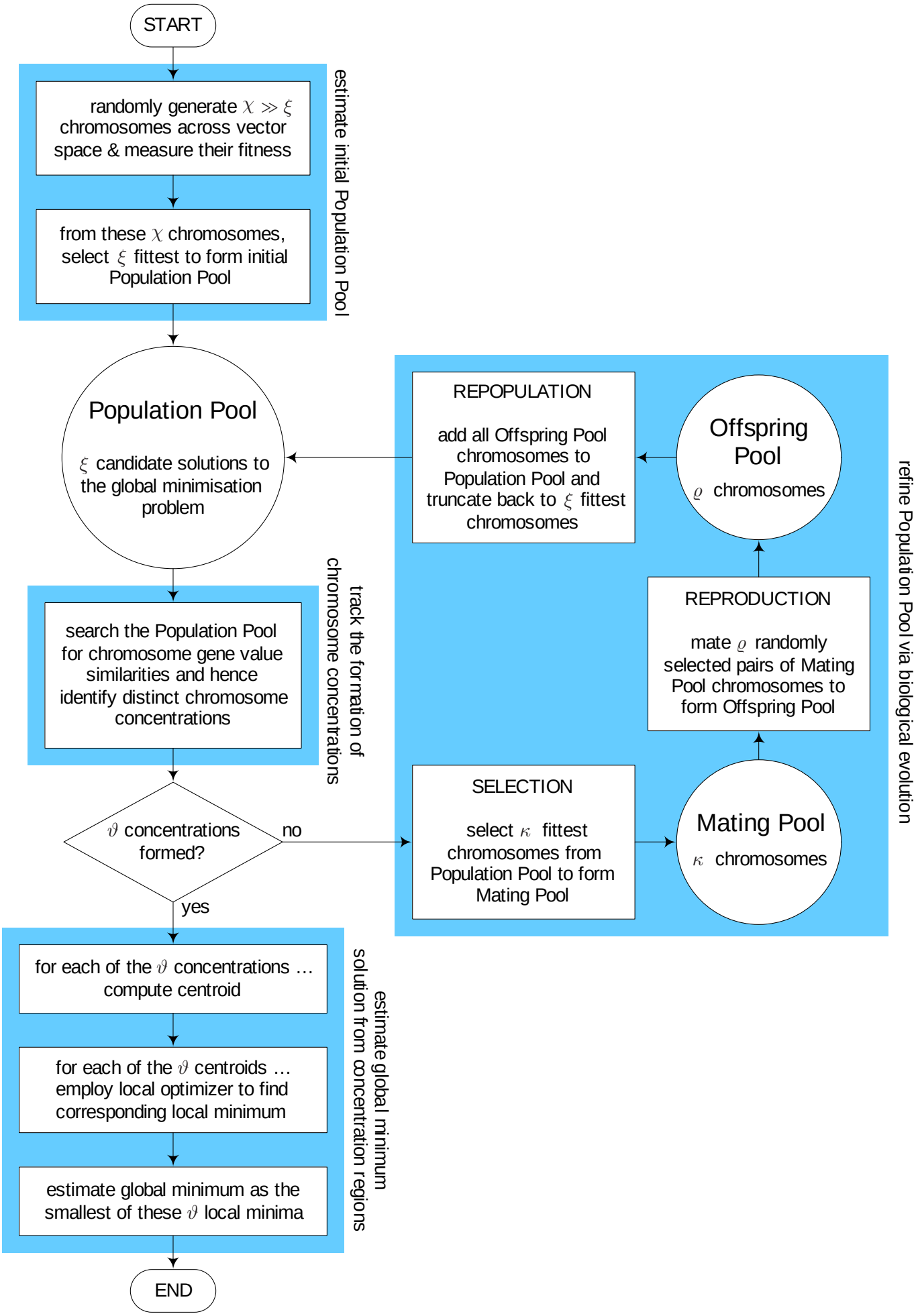


Figure A.10: Flowchart of Genetic optimization algorithm

Appendix B

Mathematical Derivations & Formulas

B.1 Estimation of the Gradient Vector $\boldsymbol{g}$

B.1.1 Finite Differences

In cases when the objective function $B(\boldsymbol{h})$ exhibits minimal randomness, the Gradient vector $\boldsymbol{g}$:

$$\boldsymbol{g} \triangleq \left.\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}}\right|_{\boldsymbol{h}_k} = \left.\begin{bmatrix} \frac{\partial B(\boldsymbol{h})}{\partial h_1} & \frac{\partial B(\boldsymbol{h})}{\partial h_2} & \cdots & \frac{\partial B(\boldsymbol{h})}{\partial h_n} \end{bmatrix}^T\right|_{\boldsymbol{h}_k} \tag{B.1}$$

can be estimated numerically via the traditional *Finite Differences* technique. In the *forward-difference* version of the technique, the i^{th} element of $\boldsymbol{g}$ is estimated as:

$$\left.\frac{\partial B(\boldsymbol{h})}{\partial h_i}\right|_{\boldsymbol{h}_k} \approx \frac{B(\boldsymbol{h}_k + \epsilon\, \boldsymbol{e}_i) - B(\boldsymbol{h}_k)}{\epsilon} \tag{B.2}$$

where ϵ is a small positive scalar and $\boldsymbol{e}_i$ is the unit vector along the i^{th} dimension of the vector space $\boldsymbol{h}$. In total, $(n+1)$ objective function measurements are required to compute this estimate.

In the more accurate *central-difference* version of the technique, the i^{th} element of $\boldsymbol{g}$ is estimated as:

$$\left.\frac{\partial B(\boldsymbol{h})}{\partial h_i}\right|_{\boldsymbol{h}_k} \approx \frac{B(\boldsymbol{h}_k + \epsilon\, \boldsymbol{e}_i) - B(\boldsymbol{h}_k - \epsilon\, \boldsymbol{e}_i)}{2\epsilon} \tag{B.3}$$

With this more accurate estimate comes the need for a greater number of objective function measurements, specifically $2n$ in total. In both versions of the technique,

ϵ is chosen as small as possible without becoming affected by the finite arithmetic errors introduced by the computer [54].

B.1.2 Least Squares

In cases when the objective function $B(\boldsymbol{h})$ exhibits substantial randomness (measurement noise, stochastic variance), the previous *Finite Differences* estimation technique becomes unreliable and must be abandoned for this more robust but less efficient *Least Squares* estimation technique. It is noted that this technique is not borrowed from the literature but rather developed here in direct response to the needs of this research.

Let the objective function in the vicinity of $\boldsymbol{h}_k$ be estimated in terms of a linear parametric model as follows:

$$B(\,\boldsymbol{h}_k + \boldsymbol{d}\,) \;\approx\; c + \boldsymbol{d}^T\boldsymbol{a} \tag{B.4}$$

Given a set of m objective function measurements taken uniformly within the vicinity of $\boldsymbol{h}_k$:

$$\left\{\; B(\,\boldsymbol{h}_k + \boldsymbol{d}_1\,),\; B(\,\boldsymbol{h}_k + \boldsymbol{d}_2\,),\; \cdots,\; B(\,\boldsymbol{h}_k + \boldsymbol{d}_m\,) \;\right\} \tag{B.5}$$

the goal is to compute the parameters c and $\boldsymbol{a}$ of the linear model so as to minimize the mean squared estimation error. That is, c and $\boldsymbol{a}$ are computed as:

$$\underset{c,\,\boldsymbol{a}}{\arg\min}\left(\;\underset{i}{\mathrm{Av}}\left[\;\left\{\,B(\,\boldsymbol{h}_k + \boldsymbol{d}_i\,) - \left(c + \boldsymbol{d}_i^T\boldsymbol{a}\right)\right\}^2\;\right]\;\right) \tag{B.6}$$

where $\mathrm{Av}[\,\cdot\,]$ is the average operator. This specific value of the parameter $\boldsymbol{a}$ is then taken to be the desired estimate of the objective function Gradient vector $\boldsymbol{g}$. Three points are worth noting:

- Quality of the Gradient vector estimate is directly determined by the choice of objective function measurement points in Equation (B.5). Highest quality is achieved when points are chosen **uniformly** from within a **small** boxed region $[\,(\boldsymbol{h}_k - \boldsymbol{\epsilon}) \le \boldsymbol{h} \le (\boldsymbol{h}_k + \boldsymbol{\epsilon})\,]$ surrounding $\boldsymbol{h}_k$. In this context, one strategy is to partition the small boxed region into a uniform grid of measurement points. An alternative, less structured strategy is to perform a large number of random selections from within the small boxed region, yielding a near-uniform distribution of measurement points. A hybrid of the above two strategies could also be used with good effect.

  In all of the above strategies for choosing objective function measurement points, the number of points chosen should be substantial. The greater the number of measurement points, the more robust this *Least Squares* estimation

technique is against objective function randomness. A safe recommendation is at least $3n$ points.

- Since the mean squared estimation error of (B.6) is derived from the set of objective function measurement points in (B.5), it is more appropriate to discuss this Least Squares technique in terms of set averages $\mathrm{Av}[\,\cdot\,]$ rather than probabilistic expectations $E[\,\cdot\,]$. Irrespective, the mathematical theory is identical in both cases.

- To obtain the optimal linear model in the presence of substantial objective function randomness, both parameters c and $\boldsymbol{a}$ must be included in the minimization of (B.6). Choosing c deterministically, instead of including it in the minimization, leads to a degraded estimate of $\boldsymbol{g}$.

We begin by rewriting the linear parametric model of (B.4) more compactly as:

$$c + \boldsymbol{d}^T\boldsymbol{a} \;=\; \hat{\boldsymbol{d}}^T\boldsymbol{w} \tag{B.7}$$

where:

$$\boldsymbol{d} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} \qquad \boldsymbol{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \qquad \hat{\boldsymbol{d}} = \begin{bmatrix} \boldsymbol{d} \\ 1 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \\ 1 \end{bmatrix} \qquad \boldsymbol{w} = \begin{bmatrix} \boldsymbol{a} \\ c \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \\ c \end{bmatrix} \tag{B.8}$$

Substituting this compact form into (B.6) and replacing the 'arg min' operator subscripts $(c,\, \boldsymbol{a})$ with $\boldsymbol{w}$ gives:

$$\underset{\boldsymbol{w}}{\arg\min} \left(\underset{i}{\mathrm{Av}} \left[\left\{ B(\,\boldsymbol{h}_k + \boldsymbol{d}_i\,) - \hat{\boldsymbol{d}}_i^T\boldsymbol{w} \right\}^2 \right] \right) \tag{B.9}$$

Expanding the inner squared term of (B.9) gives:

$$\underset{\boldsymbol{w}}{\arg\min} \left(\underset{i}{\mathrm{Av}} \left[B(\,\boldsymbol{h}_k + \boldsymbol{d}_i\,)^2 \;+\; \boldsymbol{w}^T\hat{\boldsymbol{d}}_i\,\hat{\boldsymbol{d}}_i^T\boldsymbol{w} \;-\; 2\,B(\,\boldsymbol{h}_k + \boldsymbol{d}_i\,)\,\hat{\boldsymbol{d}}_i^T\boldsymbol{w} \right] \right) \tag{B.10}$$

From the linearity property of the $\mathrm{Av}[\,\cdot\,]$ operator, (B.10) can be rewritten as:

$$\underset{\boldsymbol{w}}{\arg\min} \left(\underset{i}{\mathrm{Av}} \left[B(\,\boldsymbol{h}_k + \boldsymbol{d}_i\,)^2 \right] + \boldsymbol{w}^T \underset{i}{\mathrm{Av}} \left[\hat{\boldsymbol{d}}_i\,\hat{\boldsymbol{d}}_i^T \right] \boldsymbol{w} \;-\; 2\,\underset{i}{\mathrm{Av}} \left[B(\,\boldsymbol{h}_k + \boldsymbol{d}_i\,)\,\hat{\boldsymbol{d}}_i^T \right] \boldsymbol{w} \right) \tag{B.11}$$

Let:

$$C = \underset{i}{\text{Av}}\left[B(\,\boldsymbol{h}_k + \boldsymbol{d}_i\,)^2 \right] \tag{B.12}$$

$$\boldsymbol{R} = \underset{i}{\text{Av}}\left[\hat{\boldsymbol{d}}_i\, \hat{\boldsymbol{d}}_i^{\,T} \right] \tag{B.13}$$

$$\boldsymbol{P} = \underset{i}{\text{Av}}\left[B(\,\boldsymbol{h}_k + \boldsymbol{d}_i\,)\, \hat{\boldsymbol{d}}_i^{\,T} \right] \tag{B.14}$$

(B.11) can then be rewritten compactly as:

$$\underset{\boldsymbol{w}}{\arg\min}\left(\; C \,+\, \boldsymbol{w}^T\boldsymbol{R}\,\boldsymbol{w} \,-\, 2\boldsymbol{P}\,\boldsymbol{w} \; \right) \tag{B.15}$$

The bracketed term of (B.15) is recognized as a convex quadratic function of $\boldsymbol{w}$. Convexity results from the fact that $\boldsymbol{R}$ is positive-definite; that is, given a set of $\boldsymbol{d}_i$ chosen **uniformly** in the vicinity of $\boldsymbol{h}_k$:

$$\boldsymbol{w}^T\boldsymbol{R}\,\boldsymbol{w} = \boldsymbol{w}^T \underset{i}{\text{Av}}\left[\hat{\boldsymbol{d}}_i\, \hat{\boldsymbol{d}}_i^{\,T} \right] \boldsymbol{w} \tag{B.16}$$

$$= \underset{i}{\text{Av}}\left[\boldsymbol{w}^T\hat{\boldsymbol{d}}_i\, \hat{\boldsymbol{d}}_i^{\,T}\boldsymbol{w} \right] \tag{B.17}$$

$$= \underset{i}{\text{Av}}\left[\boldsymbol{w}^T\hat{\boldsymbol{d}}_i\, \boldsymbol{w}^T\hat{\boldsymbol{d}}_i \right] \tag{B.18}$$

$$= \underset{i}{\text{Av}}\left[\left(\boldsymbol{w}^T\hat{\boldsymbol{d}}_i \right)^2 \right] \;>\; 0 \;\forall\; \boldsymbol{w} \neq \boldsymbol{0} \tag{B.19}$$

It follows that (B.15) has a unique solution of $\boldsymbol{w}$ existing where the Gradient of the quadratic function equals zero:

$$\frac{\partial}{\partial \boldsymbol{w}}\left(\; C \,+\, \boldsymbol{w}^T\boldsymbol{R}\,\boldsymbol{w} \,-\, 2\boldsymbol{P}\,\boldsymbol{w} \; \right) = \boldsymbol{0} \tag{B.20}$$

$$2\boldsymbol{R}\boldsymbol{w} - 2\boldsymbol{P} = \boldsymbol{0} \tag{B.21}$$

$$\boldsymbol{R}\boldsymbol{w} = \boldsymbol{P} \tag{B.22}$$

$$\boldsymbol{w} = \boldsymbol{R}^{-1}\boldsymbol{P} \tag{B.23}$$

Mindful of matrix conditioning effects, $\boldsymbol{w}$ is computed via the (B.22) linear system of equations (LU decomposition of $\boldsymbol{R}$) instead of the (B.23) $\boldsymbol{R}$ inversion [213]. From the definition of $\boldsymbol{w}$ given in (B.8), the desired $\boldsymbol{a}$ and therefore $\boldsymbol{g}$ estimate is simply the first n elements of $\boldsymbol{w}$.

B.2 Estimation of the Hessian Matrix $\boldsymbol{H}$

B.2.1 Finite Differences

In cases when the objective function $B(\boldsymbol{h})$ exhibits minimal randomness, the Hessian matrix $\boldsymbol{H}$:

$$\boldsymbol{H} \triangleq \frac{\partial}{\partial \boldsymbol{h}} \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \Bigg|_{\boldsymbol{h}_k} = \left[\begin{array}{cccc} \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1 \, \partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1 \, \partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1 \, \partial h_n} \\ \\ \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2 \, \partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2 \, \partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2 \, \partial h_n} \\ \\ \vdots & \vdots & \ddots & \vdots \\ \\ \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n \, \partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n \, \partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n \, \partial h_n} \end{array} \right] \Bigg|_{\boldsymbol{h}_k} \tag{B.24}$$

can be estimated numerically via the same *Finite Differences* concepts first introduced in Appendix B.1.1 for estimating the Gradient vector $\boldsymbol{g}$. Knowledge of these concepts is assumed in the following discussion.

Let the $(i,j)^{\text{th}}$ element of $\boldsymbol{H}$ be rewritten as:

$$\frac{\partial^2 B(\boldsymbol{h})}{\partial h_i \, \partial h_j} \Bigg|_{\boldsymbol{h}_k} = \left[\frac{\partial}{\partial h_i} \left(\frac{\partial B(\boldsymbol{h})}{\partial h_j} \right) \right] \Bigg|_{\boldsymbol{h}_k} \tag{B.25}$$

If the round bracketed term of (B.25) is interpreted simply as a function of $\boldsymbol{h}$ then the right hand side can be estimated in terms of *forward differences*:

$$\frac{\partial^2 B(\boldsymbol{h})}{\partial h_i \, \partial h_j} \Bigg|_{\boldsymbol{h}_k} \approx \frac{\left(\frac{\partial B(\boldsymbol{h})}{\partial h_j} \right) \Big|_{\boldsymbol{h}_k + \epsilon \boldsymbol{e}_i} - \left(\frac{\partial B(\boldsymbol{h})}{\partial h_j} \right) \Big|_{\boldsymbol{h}_k}}{\epsilon} \tag{B.26}$$

where ϵ is a small positive scalar and $\boldsymbol{e}_i$ is the unit vector along the i^{th} dimension of the vector space $\boldsymbol{h}$. It is worth noting that a more accurate *central difference* estimate could be used here however practical implementations generally favor the *forward difference* estimate due to its greater efficiency in the long run.

Estimating the numerator partial derivatives of (B.26), again in terms of *forward differences*, gives:

$$\left.\frac{\partial^2 B(\boldsymbol{h})}{\partial h_i\,\partial h_j}\right|_{\boldsymbol{h}_k} \approx \frac{\left(\dfrac{B(\boldsymbol{h}_k+\epsilon\boldsymbol{e}_i+\epsilon\boldsymbol{e}_j)-B(\boldsymbol{h}_k+\epsilon\boldsymbol{e}_i)}{\epsilon}\right)-\left(\dfrac{B(\boldsymbol{h}_k+\epsilon\boldsymbol{e}_j)-B(\boldsymbol{h}_k)}{\epsilon}\right)}{\epsilon} \tag{B.27}$$

Rearranging (B.27) then gives the desired estimate of the $(i,j)^{\text{th}}$ element of $\boldsymbol{H}$ in terms of objective function measurements:

$$\left.\frac{\partial^2 B(\boldsymbol{h})}{\partial h_i\,\partial h_j}\right|_{\boldsymbol{h}_k} \approx \frac{B(\boldsymbol{h}_k+\epsilon\boldsymbol{e}_i+\epsilon\boldsymbol{e}_j)-B(\boldsymbol{h}_k+\epsilon\boldsymbol{e}_i)-B(\boldsymbol{h}_k+\epsilon\boldsymbol{e}_j)+B(\boldsymbol{h}_k)}{\epsilon^2} \tag{B.28}$$

If the objective function measurements in (B.28) are stored in memory once acquired and appropriately recycled for all elements of the Hessian matrix, a total of $\binom{n^2-n}{2}+2n+1$ objective function measurements are required to form the $\boldsymbol{H}$ estimate.

B.2.2 Hybridization of Finite Differences & Least Squares

In cases when the objective function $B(\boldsymbol{h})$ exhibits substantial randomness (measurement noise, stochastic variance), partial derivatives of $B(\boldsymbol{h})$ cannot be reliably estimated via *Finite Differences*. In the context of the previous section, this means the step from (B.26) to (B.27) falls down and leads to a generally poor estimate in (B.28).

This problem can be avoided altogether however by row vectorizing the estimation of $\boldsymbol{H}$ (estimating $\boldsymbol{H}$ row-by-row instead of element-by-element), in which case partial derivatives of $B(\boldsymbol{h})$ no longer appear individually but rather are grouped into Gradient vectors which can be robustly estimated by the *Least Squares* technique of Appendix B.1.2. The Hessian estimation technique thus born and presented here is more robust in the presence of objective function randomness but expectedly less efficient.

As will become evident below, this robust estimation technique can be considered a hybrid technique since rows of $\boldsymbol{H}$ are initially estimated in terms of *Finite Differences* whilst subsequent partial derivatives of $B(\boldsymbol{h})$ are estimated in terms of *Least Squares*.

With the Hessian matrix $\boldsymbol{H}$ defined as:

$$\boldsymbol{H} \triangleq \frac{\partial}{\partial \boldsymbol{h}} \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \Bigg|_{\boldsymbol{h}_k} = \left[\begin{array}{cccc} \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1 \, \partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1 \, \partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_1 \, \partial h_n} \\ \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2 \, \partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2 \, \partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_2 \, \partial h_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n \, \partial h_1} & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n \, \partial h_2} & \cdots & \frac{\partial^2 B(\boldsymbol{h})}{\partial h_n \, \partial h_n} \end{array} \right] \Bigg|_{\boldsymbol{h}_k} \tag{B.29}$$

the i^{th} row of $\boldsymbol{H}$ can be expressed as:

$$\boldsymbol{H}_{i,\forall} = \frac{\partial}{\partial h_i} \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \Bigg|_{\boldsymbol{h}_k} \tag{B.30}$$

Estimating the outer partial derivative of (B.30) via *forward differences* then gives:

$$\boldsymbol{H}_{i,\forall} = \frac{\left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \Big|_{\boldsymbol{h}_k + \epsilon \boldsymbol{e}_i} - \left(\frac{\partial B(\boldsymbol{h})}{\partial \boldsymbol{h}} \right)^T \Big|_{\boldsymbol{h}_k}}{\epsilon} \tag{B.31}$$

where ϵ is a small positive scalar and $\boldsymbol{e}_i$ is the unit vector along the i^{th} dimension of the vector space $\boldsymbol{h}$. It is worth noting that (B.30) and (B.31) are the row vectorized extension of (B.25) and (B.26) respectively of the previous section.

Since the left and right numerator terms in (B.31) are recognized as the Gradient vector of $B(\boldsymbol{h})$ at $(\boldsymbol{h}_k + \epsilon \boldsymbol{e}_i)$ and $\boldsymbol{h}_k$ respectively, each can be estimated via the robust *Least Squares* technique of Appendix B.1.2. With Appendix B.1.2 recommending at least $3n$ objective function measurements per Gradient estimate, it follows that a total of at least $3n^2 + 3n$ objective function measurements are required to form the entire $\boldsymbol{H}$ estimate, assuming each Gradient estimate is computed independently and the right numerator term of (B.31) is recycled for each row.

B.3 Eigen Properties of the Hessian Matrix $\boldsymbol{H}$

Presented below are the eigen properties of the Hessian matrix $\boldsymbol{H}$ that are specifically utilized in the theory of the *Trust Region Newton* and *Alpha Branch & Bound* optimization algorithms:

Property 1: Since $\boldsymbol{H}$ is a square $(n \times n)$, real, symmetric matrix [118]:

- all eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ are real
- a full set of n linearly independent and orthonormal eigenvectors $\boldsymbol{q}_1, \boldsymbol{q}_2, \cdots, \boldsymbol{q}_n$ exist whereby $\boldsymbol{q}_i^T \boldsymbol{q}_j = 0$ for $i \neq j$ and $\boldsymbol{q}_i^T \boldsymbol{q}_j = 1$ for $i = j$

Property 2: As a consequence of Property 1, $\boldsymbol{H}$ can be written in diagonal form [11] as:

$$\boldsymbol{H} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T \tag{B.32}$$

where $\boldsymbol{\Lambda}$ is the diagonal *spectral* matrix:

$$\boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \tag{B.33}$$

and $\boldsymbol{Q}$ is the orthogonal *modal* matrix:

$$\boldsymbol{Q} = [\boldsymbol{q_1}\ \boldsymbol{q_2}\ \cdots\ \boldsymbol{q_n}] \tag{B.34}$$

The orthogonality of $\boldsymbol{Q}$ stems from the fact [143]:

$$\boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I} = \boldsymbol{Q}^{-1}\boldsymbol{Q} \quad \text{and therefore} \quad \boldsymbol{Q}^T = \boldsymbol{Q}^{-1} \tag{B.35}$$

Property 3: Geometrically, the eigenvectors of $\boldsymbol{H}$ define the principle axes of the quadratic surface $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d}$ whilst the eigenvalues of $\boldsymbol{H}$ represent the second order derivatives of $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d}$ with respect to the principle axes [285]. A principle axis of $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d}$ is defined as the set of linear points $C\boldsymbol{q}$ for which the gradient is collinear with the origin. A principle axis can be visualized as passing orthogonally through the hyper-ellipsoidal contours of the $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d}$ surface. This same geometrical interpretation can be applied to the full quadratic model of (A.10), only this time the quadratic surface will be translated in the $\boldsymbol{d}$ vector space by the inclusion of the constant and linear terms.

Property 4: It can be shown mathematically [118] or interpreted geometrically from Property 3 (eigenvalues represent second order derivatives) that the eigenvalues of $\boldsymbol{H}$ determine the *definiteness* of the quadratic term $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d}$:

- If all eigenvalues of $\boldsymbol{H}$ are positive than $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d} > 0$ for all $\boldsymbol{d} \neq \boldsymbol{0}$ and $\boldsymbol{H}$ is called *positive-definite*.
- If all eigenvalues of $\boldsymbol{H}$ are non-negative and at least one of the eigenvalues is zero then $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d} \geq 0$ for all $\boldsymbol{d} \neq \boldsymbol{0}$ and $\boldsymbol{H}$ is called *positive-semidefinite*.
- If all eigenvalues of $\boldsymbol{H}$ are negative than $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d} < 0$ for all $\boldsymbol{d} \neq \boldsymbol{0}$ and $\boldsymbol{H}$ is called *negative-definite*.
- If all eigenvalues of $\boldsymbol{H}$ are non-positive and at least one of the eigenvalues is zero then $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d} \leq 0$ for all $\boldsymbol{d} \neq \boldsymbol{0}$ and $\boldsymbol{H}$ is called *negative-semidefinite*.
- If the eigenvalues of $\boldsymbol{H}$ are both positive and negative than $\boldsymbol{d}^T\boldsymbol{H}\boldsymbol{d}$ takes at least one positive value and at least one negative value and $\boldsymbol{H}$ is called *indefinite*.

Property 5: The eigenvalues of $[\boldsymbol{H} + \lambda\boldsymbol{I}]$ are $(\lambda_1 + \lambda, \lambda_2 + \lambda, \cdots, \lambda_n + \lambda)$ whilst the eigenvectors of $[\boldsymbol{H} + \lambda\boldsymbol{I}]$ are the same as the eigenvectors of $\boldsymbol{H}$ [275]. It follows that weighting the entire diagonal of $\boldsymbol{H}$ by a constant simply changes the second order characteristics along the principle axes.

Property 6: The determinant of $\boldsymbol{H}$ can be derived from its eigenvalues:

$$\det(\boldsymbol{H}) = \prod_{i=1}^{n} \lambda_i \tag{B.36}$$

If $\det(\boldsymbol{H}) \neq 0$ then $\text{rank}(\boldsymbol{H}) = n$ and $\boldsymbol{H}$ is nonsingular. In this case the linear system of equations $\boldsymbol{H}\boldsymbol{x} = \boldsymbol{b}$ has a unique solution.

If $\det(\boldsymbol{H}) = 0$ then $\text{rank}(\boldsymbol{H}) < n$ and $\boldsymbol{H}$ is singular. In this case, the linear system of equations $\boldsymbol{H}\boldsymbol{x} = \boldsymbol{b}$ has either a non-unique solution or no solution exists.

Property 7: Let R_g be a general boxed region of vector space $\boldsymbol{h}$ with domain:

$$\left[\boldsymbol{h}^{L,R_g} \leq \boldsymbol{h} \leq \boldsymbol{h}^{U,R_g}\right] \tag{B.37}$$

where $\boldsymbol{h}^{L,R_g}$ and $\boldsymbol{h}^{U,R_g}$ represent the lower and upper constraints of region R_g respectively. Within region R_g, it is assumed that the Hessian matrix of the function of $\boldsymbol{h}$ is itself dependent on $\boldsymbol{h}$ and therefore represented as:

$$\boldsymbol{H}(\boldsymbol{h}) = \begin{bmatrix} H_{1,1}(\boldsymbol{h}) & H_{1,2}(\boldsymbol{h}) & \cdots & H_{1,n}(\boldsymbol{h}) \\ H_{2,1}(\boldsymbol{h}) & H_{2,2}(\boldsymbol{h}) & \cdots & H_{2,n}(\boldsymbol{h}) \\ \vdots & \vdots & \ddots & \vdots \\ H_{n,1}(\boldsymbol{h}) & H_{n,2}(\boldsymbol{h}) & \cdots & H_{n,n}(\boldsymbol{h}) \end{bmatrix} \tag{B.38}$$

If:

$$\bar{H}_{i,j} = \max_{\boldsymbol{h}^{L,R_g} \leq \boldsymbol{h} \leq \boldsymbol{h}^{U,R_g}} H_{i,j}(\boldsymbol{h}) \tag{B.39}$$

and

$$\underline{H}_{i,j} = \min_{\boldsymbol{h}^{L,R_g} \leq \boldsymbol{h} \leq \boldsymbol{h}^{U,R_g}} H_{i,j}(\boldsymbol{h}) \tag{B.40}$$

then the interval Hessian matrix of region R_g is defined as:

$$[\boldsymbol{H}]^{R_g} = \begin{bmatrix} \left[\underline{H}_{1,1}, \bar{H}_{1,1}\right] & \left[\underline{H}_{1,2}, \bar{H}_{1,2}\right] & \cdots & \left[\underline{H}_{1,n}, \bar{H}_{1,n}\right] \\ \left[\underline{H}_{2,1}, \bar{H}_{2,1}\right] & \left[\underline{H}_{2,2}, \bar{H}_{2,2}\right] & \cdots & \left[\underline{H}_{2,n}, \bar{H}_{2,n}\right] \\ \vdots & \vdots & \ddots & \vdots \\ \left[\underline{H}_{n,1}, \bar{H}_{n,1}\right] & \left[\underline{H}_{n,2}, \bar{H}_{n,2}\right] & \cdots & \left[\underline{H}_{n,n}, \bar{H}_{n,n}\right] \end{bmatrix} \tag{B.41}$$

where $\left[\underline{H}_{i,j}, \bar{H}_{i,j}\right]$ represents the closed interval $\underline{H}_{i,j} \leq H_{i,j} \leq \bar{H}_{i,j}$. (B.41) can be written in shorthand as:

$$[\boldsymbol{H}]^{R_g} = \left[\underline{\boldsymbol{H}}, \bar{\boldsymbol{H}}\right] \tag{B.42}$$

where:

$$\underline{\boldsymbol{H}} = \begin{bmatrix} \underline{H}_{1,1} & \underline{H}_{1,2} & \cdots & \underline{H}_{1,n} \\ \underline{H}_{2,1} & \underline{H}_{2,2} & \cdots & \underline{H}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \underline{H}_{n,1} & \underline{H}_{n,2} & \cdots & \underline{H}_{n,n} \end{bmatrix} \tag{B.43}$$

and

$$\bar{\boldsymbol{H}} = \begin{bmatrix} \bar{H}_{1,1} & \bar{H}_{1,2} & \cdots & \bar{H}_{1,n} \\ \bar{H}_{2,1} & \bar{H}_{2,2} & \cdots & \bar{H}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{H}_{n,1} & \bar{H}_{n,2} & \cdots & \bar{H}_{n,n} \end{bmatrix} \tag{B.44}$$

The entire set of matrices represented by the interval bounds of $[\boldsymbol{H}]^{R_g}$ is referred to as the $[\boldsymbol{H}]^{R_g}$ interval Hessian matrix family $\left\{ [\boldsymbol{H}]^{R_g} \right\}$. By definition:

$$\left\{ \boldsymbol{H}(\boldsymbol{h}) \ \forall \ \left[\boldsymbol{h}^{L,R_g} \leq \boldsymbol{h} \leq \boldsymbol{h}^{U,R_g} \right] \right\} \ \subseteq \ \left\{ [\boldsymbol{H}]^{R_g} \right\} \tag{B.45}$$

Property 8: Let $\boldsymbol{H}$ be the arbitrary (not necessarily Hessian) square matrix:

$$\boldsymbol{H} = \begin{bmatrix} H_{1,1} & H_{1,2} & \cdots & H_{1,n} \\ H_{2,1} & H_{2,2} & \cdots & H_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ H_{n,1} & H_{n,2} & \cdots & H_{n,n} \end{bmatrix} \tag{B.46}$$

From matrix $\boldsymbol{H}$, n complex-z-plane disks D_i can be defined as:

$$|\, z - H_{i,i} \,| \ \leq \ \sum_{\substack{j=1 \\ j \neq i}}^{n} |\, H_{i,j} \,| \quad \text{for integer} \quad 1 \leq i \leq n \tag{B.47}$$

It is seen that each disk D_i is centered at diagonal element $H_{i,i}$ with radius equal to the sum of the magnitudes of all row i off-diagonal elements. Gerschgorin's Theorem [118] then states that each disk D_i represents the entire bounds of one of the eigenvalues of $\boldsymbol{H}$; n disks, n eigenvalues. It is noted that in this theorem, each eigenvalue is counted with its algebraic multiplicity.

For the special case when $\boldsymbol{H}$ is a real symmetric matrix (Hessian), all eigenvalues must be real according to the previous Property 1. It follows that the complex bounding region of each disk can be reduced to a real bounding interval equal to that part of the real number line enclosed by the disk. It is worth noting that since the matrix is real in this case, the center of each disk $(H_{i,i})$ will lie on the real number line.

From the above discussion, it logically follows that diagonally weighting a square matrix has the effect of translating its Gerschgorin disks and therefore its eigenvalue bounds.

Appendix C

Patent

The full document set representing the patent description, drawings, claims, application filings and international search report is included on the accompanying DVD within the *Patent* folder. For brevity, only patent claims are included in this appendix. These claims represent the scope of the IP generated from this research.

CLAIMS

1. A method for linearising a multi-carrier radio frequency transmitter or a multi-user CDMA radio frequency transmitter, including the steps of:

measuring a function of out-of-band signal power in the frequency domain at an output of the radio frequency transmitter; and

applying digital base-band pre-distortion to the radio frequency transmitter according to the measured function of the out-of-band signal power;

wherein the digital base-band pre-distortion is performed by a digital base-band pre-distortion network.

2. The method of claim 1 wherein digital base-band pre-distortion network coefficients of the digital base-band pre-distortion network are optimised to minimise the measured function of the out-of-band signal power.

3. The method of claim 2 wherein the digital base-band pre-distortion network coefficients are optimised whilst the transmitter is broadcasting.

4. The method of claim 1 wherein the digital base-band pre-distortion network is a non-linear behavioural model with memory.

5. The method of claim 4 wherein the non-linear behavioural model with memory is a pruned Volterra Series.

6. The method of claim 2 wherein the digital base-band pre-distortion network coefficients are pruned Volterra Series kernel coefficients.

7. The method of claim 1 wherein the digital base-band pre-distortion network is given by the equation:

$$y[n] = x[n] + \sum_{a=1}^{\left(\frac{P-1}{2}\right)} \left[\sum_{k=0}^{\left\lceil \frac{M+1}{R} \right\rceil - 1} h_{2a+1}[k]\, x[n] \left|x[n]\right|^{2(a-1)} \left|x[n-Rk]\right|^{2} \right]$$

where $h_{2a+1}[k]$ are the digital base-band pre-distortion network kernel coefficients.

8. The method of claim 7 wherein the memory length M is estimated by:

a) pruning the digital base-band pre-distortion network to a 3rd order single delay digital base-band pre-distortion network given by the equation:

$$y[n] = x[n] + h_3[k]\, x[n] \left|x[n-k]\right|^2$$

b) Sweeping a delay variable (k) of the 3rd order single delay pre-distortion network from zero upwards; and

c) Observing a value of k when an asymmetry of the transmitter output adjacent channel power spectrum changes wherein the value of k is equal to the memory length M.

9. The method of claim 1 wherein the function of the out-of-band signal power is a measure of transmitter output non-linearity.

10. The method of claim 9 wherein the function of the out-of-band signal power involves accumulating a weighted out-of-band power spectral density with respect to frequency.

11. The method of claim 10 wherein the function of the out-of-band signal power is given by the equation:

$$WACP = \sum_{\text{LAC } f} W(f) \times PSD(f) + \sum_{\text{UAC } f} W(f) \times PSD(f)$$

12. The method of claim 11 wherein the weighting function $W(f)$, for either the lower adjacent channel (LAC) or upper adjacent channel (UAC), is a non-increasing function of $|f - f_I|$.

13. The method of claim 10 or claim 11 wherein the power spectral density is measured with a spectrum analyser.

14. The method of claim 7 wherein a subset of the digital base-band pre-distortion network kernel coefficients is optimised separately.

15. The method of claim 14 wherein a combination of 3rd order, a combination of 3rd and 5th order or a combination of 3rd and 5th and 7th order digital base-band pre-distortion network kernel coefficients is optimised separately.

16. The method of claim 14 wherein the digital base-band pre-distortion network kernel coefficients are optimised according to a local minimum non-gradient based algorithm.

17. The method of claim 14 wherein the digital base-band pre-distortion network kernel coefficients are optimised according to a global minimum non-gradient based algorithm.

18. The method of claim 16 wherein the local minimum non-gradient based algorithm is a Nelder-Mead Simplex algorithm.

19. The method of claim 17 wherein the global minimum non-gradient based algorithm is a Genetic algorithm.

20. The method of claim 14 wherein a subset of the digital base-band pre-distortion network kernel coefficients, all of the same non-linear order, is optimised separately according to a gradient based algorithm.

21. The method of claim 20 wherein the gradient based algorithm is a local minimum Gradient Descent algorithm.

Appendix D

Publications

The following journal articles are products of this research activity and are included in this appendix in respective order:

B. D. Laki and C. J. Kikkert, "Adaptive Digital Predistortion For Wideband High Crest Factor Applications Based On The WACP Optimization Objective: A Conceptual Overview," *IEEE Transactions On Broadcasting*, Vol. 58, No. 4, pp. 609–618, December 2012.

B. D. Laki and C. J. Kikkert, "Adaptive Digital Predistortion For Wideband High Crest Factor Applications Based On The WACP Optimization Objective: An Extended Analysis," *IEEE Transactions On Broadcasting*, Vol. 59, No. 1, pp. 136–145, March 2013.

Adaptive Digital Predistortion for Wideband High Crest Factor Applications Based on the WACP Optimization Objective: A Conceptual Overview

Bradley Dean Laki, *Graduate Member, IEEE,* and Cornelis Jan Kikkert, *Life Senior Member, IEEE*

***Abstract*—This paper proposes a method of digital predistortion suitable for wideband high crest factor applications such as those encountered in DAB, DVB-T and WCDMA transmitters. The proposed method is advantageous for four main reasons. Firstly, it utilizes a reliable frequency domain measure of transmitter output nonlinearity, specifically the Weighted Adjacent Channel Power (WACP), as the objective for predistortion filter parameter estimation. This is in direct contrast to traditional approaches which utilize a time domain measure obtained via a full feedback path and potentially corrupted by gain and phase compensation error as well as ADC distortion. Secondly, the method models predistortion filter parameter estimation as a generic nonlinear mathematical optimisation problem. This model assumes a nonconvex objective function and therefore utilizes both global and local optimization algorithms to achieve true global convergence. This is once again in direct contrast to traditional approaches which model predistortion filter parameter estimation as a linear regression problem. Such a model incorrectly assumes a convex error surface and therefore restricts itself to inadequate local optimisation algorithms which unfortunately cannot guarantee true global convergence. Thirdly, the method's predistortion filter is a pruned Volterra Series with memory which utilizes a hybrid pruning strategy in order to keep high order kernels to a practically manageable size, suitable for optimization parameter estimation. Predistortion filter memory ultimately makes the method highly suited to wideband applications. Finally, predistortion filter parameter estimation does not require known test signals to be injected into the transmitter and therefore the technique is on-air adaptive. This means any transmitter using this method of digital predistortion will be both on-air and optimally linearized for its entire operational life.**

Preliminary results obtained from actual hardware are presented.

***Index Terms*—Adjacent channel power, CDMA, DAB, DVB-T, Linearization techniques, Nonlinear distortion, OFDM, Optimization, Power amplifiers, Predistortion, Radio transmitters, Volterra Series**

Adaptive Digital Predistortion for Wideband High Crest Factor Applications Based on the WACP Optimization Objective: An Extended Analysis

Bradley Dean Laki, *Graduate Member, IEEE,* and Cornelis Jan Kikkert, *Life Senior Member, IEEE*

***Abstract*—This paper provides an extended analysis of the adaptive digital predistortion technique initially proposed and conceptually overviewed in [1]. This digital predistortion technique is suitable for wideband high crest factor applications (DAB, DVB-T & WCDMA high power transmitters) and overcomes the technical deficiencies of the traditional Direct Learning method. Specifically, predistortion filter parameter estimation is modeled as a generic mathematical optimization problem instead of a linear regression problem. In addition, the optimization objective is derived in the frequency rather than time domain. A hybridly pruned Volterra Series with memory is used to implement the predistortion filter. Hybrid pruning leads to a small optimization vector space whilst predistortion filter memory makes the method well suited to wideband applications. Given that predistortion filter parameter estimation does not rely on known test signals being injected into the transmitter, the method is on-air adaptive. Implementation aspects of the technique not covered in [1] but requiring extended coverage in this paper include selection of mathematical optimization algorithms, predistortion filter memory estimation, WACP weighting function application and on-air adaption performance. Results obtained from actual hardware are presented.**

***Index Terms*—Adjacent channel power, CDMA, DAB, DVB-T, Linearization techniques, Nonlinear distortion, OFDM, Optimization, Power amplifiers, Predistortion, Radio transmitters, Volterra Series**

Appendix E

Data Sheets

Data sheets for the laboratory transmitter testbed *driver* and *power* amplifiers are included in this appendix in respective order.

Surface Mount

Monolithic Amplifier DC-2 GHz

Features

- InGaP HBT microwave amplifier
- Miniature SOT-89 package
- Internally Matched to 50 Ohms
- Frequency range, DC to 2 GHz
- Output power, 15.5 dBm typ.
- Excellent package for heat dissipation, exposed metal bottom
- Low thermal resistance for high reliability
- Aqueous washable
- Protected by US Patent 6,943,629

Gali-52+

CASE STYLE: DF782
PRICE: $1.29 ea. QTY. (30)

+ RoHS compliant in accordance with EU Directive (2002/95/EC)

The +Suffix has been added in order to identify RoHS Compliance. See our web site for RoHS Compliance methodologies and qualifications.

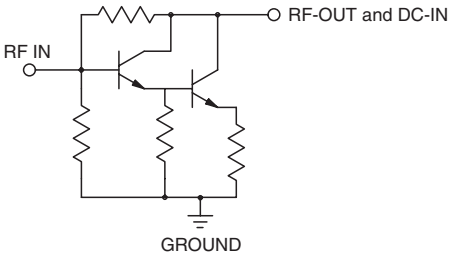
Applications

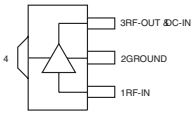
- Cellular
- PCS
- Communication receivers & transmitters

General Description

Gali-52+ (RoHS compliant) is a wideband amplifier offering high dynamic range. Lead finish is SnAgNi. It has repeatable performance from lot to lot, and is enclosed in a SOT-89 package. It uses patented Transient Protected Darlington configuration and is fabricated using InGaP HBT technology. Expected MTBF is 14,000 years at 85°C case temperature. Gali-52+ is designed to be rugged for ESD and supply switch-on transients.

simplified schematic and pin description





Function	Pin Number	Description
RF IN	1	RF input pin. This pin requires the use of an external DC blocking capacitor chosen for the frequency of operation.
RF-OUT and DC-IN	3	RF output and bias pin. DC voltage is present on this pin; therefore a DC blocking capacitor is necessary for proper operation. An RF choke is needed to feed DC bias without loss of RF signal due to the bias connection, as shown in "Recommended Application Circuit".
GND	2,4	Connections to ground. Use via holes as shown in "Suggested Layout for PCB Design" to reduce ground path inductance for best performance.

REV. R
M120653
D60129
EE-7974Q
GALI-52+
RS/YB/FL
100830

Electrical Specifications at 25°C and 50mA, unless noted

Parameter		Min.	Typ.	Max.	Units
Frequency Range*		DC		2	GHz
Gain	f=0.1 GHz		22.9		dB
	f=1 GHz		20.8		
	f=2 GHz	16	17.8		
	f=3 GHz		15.9		
	f=4 GHz		14.4		
Input Return Loss	f= DC to 2 GHz		16.5		dB
Output Return Loss	f= DC to 2 GHz		15.5		dB
Output Power @ 1 dB compression	f=1 GHz	13.5	15.5		dBm
Output IP3	f=1 GHz		32		dBm
Noise Figure	f=1 GHz		2.7		dB
Recommended Device Operating Current			50		mA
Device Operating Voltage		4.0	4.4	4.8	V
Device Voltage Variation vs. Temperature at 50 mA			-3.2		mV/°C
Device Voltage Variation vs. Current at 25°C			3.5		mV/mA
Thermal Resistance, junction-to-case[1]			85		°C/W

*Guaranteed specification DC-2 GHz. Low frequency cut off determined by external coupling capacitors.

Absolute Maximum Ratings

Parameter	Ratings
Operating Temperature*	-45°C to 85°C
Storage Temperature	-65°C to 150°C
Operating Current	65mA
Input Power	13dBm

Note: Permanent damage may occur if any of these limits are exceeded.
These ratings are not intended for continuous normal operation.
[1]Case is defined as ground leads.
*Based on typical case temperature rise 3°C above ambient.

5300 Beethoven Street, Los Angeles, CA 90066
TEL: (310)306-5556 • FAX: (310)821-7413
WEB: www.ophirrf.com • E-MAIL: sales@ophirrf.com

MODEL 5303038

20 - 1000 MHz
25 WATTS
LINEAR POWER RF AMPLIFIER

Solid State Broadband High Power RF Amplifier

The 5303038 is a 25 Watt broadband amplifier that covers the 20 – 1000 MHz frequency range. This small and lightweight amplifier utilizes Class A/AB linear power devices that provide an excellent 3rd order intercept point, high gain, and a wide dynamic range.

Due to robust engineering and employment of the most advanced devices and components, this amplifier achieves high efficiency operation with proven reliability. Like all OPHIR$_{RF}$ amplifiers, the 5303038 comes with an extended multiyear warranty.

	Parameter	Specification @ 25° C
Electrical		
1	Frequency Range	20 – 1000 MHz
2	Saturated Output Power	25 Watts typical
3	Power Output @ 1dB Comp.	10 Watts min
4	Small Signal Gain	+46 dB min
5	Gain Flatness	± 1.5 dB max
6	IP_3	+48 dBm typical
7	Input VSWR	2:1 max
8	Harmonics	-20 dBc typical @ 10 Watts
9	Spurious Signals	< -60 dBc typical @ 10 Watts
10	Input/Output Impedance	50 Ohms nominal
11	DC Input Current	4.5 Amps max
12	DC Input	28 VDC nominal
13	RF Input	0 dBm max
14	RF Input Signal Format	CW/AM/FM/PM/Pulse
15	Class of Operation	AB
Mechanical		
16	Dimensions	6” x 3” x 1.1”
17	Weight	2 lb. max
18	Connectors	SMA female
19	Grounding	Chassis
20	Cooling	Adequate Heatsink Required
Environmental		
21	Operating Temperature	0º C to +50º C
22	Operating Humidity	95% Non-condensing
23	Operating Altitude	Up to 10,000’ Above Sea Level
24	Shock and Vibration	Normal Truck Transport

Specifications subject to change without notice.

Solid State RF Amplifier with Push-Pull Circuitry

04/09 Approved By: _____ Date: _____