Linearization of Concurrent Dual-Band Power Amplifier Using Digital Predistortion

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of The Ohio State University

By

Naveen Naraharisetti, B.Tech., M.S.

Graduate Program in Electrical and Computer Engineering

The Ohio State University

2014

Dissertation Committee:

Patrick Roblin, Adviser
Joanne E. Degroat
Furrukh Khan
Cynthia A. Gerhardt
ABSTRACT

With the current demand for higher data-rate transmissions, complex envelop techniques like wideband code division multiple access (WCDMA) and orthogonal frequency division multiplexing (OFDM) techniques are employed because of their high spectral efficiency. However these modulated schemes which feature non-constant envelop signals with high peak to average power ratio (PAPR) also impose strict linearity requirements on the PAs. A digital predistortion is used to linearize the PA for its robustness and easy implementation on an FPGA. The complexity of the DPD design is alleviated using widely spaced carriers and frequency selective technique.

A new direct learning (DL) technique using 2D quasi exact inverse (2D-QEI) of a power amplifier (PA) model for linearizing concurrent dual band PA is developed. In contrast to indirect learning (IL) architecture, where the coefficients are extracted by swapping the input and output variables in any PA model, a QEI of a PA model can be used in the digital predistorter (DPD). An FPGA test bench integrating the concurrent dual band RF system is utilized to verify the linearization performance of the new DL technique. A 10W PA is excited with two WCDMA signals of 3.84MHz bandwidth which are 310 MHz apart. The measurement results demonstrate that in the presence of additive noise, there is a noticeable improvement in terms of normalized mean square error (NMSE) and adjacent channel power ratio (ACPR) when
using the QEI model for DPD. This improvement is achieved in one iteration as in practical DPD systems where data is never played twice.

To relish the advantages provided by the piece-wise polynomials, two different types of 2D cubic-spline digital predistorters are developed for linearizing a power amplifier used in dual band transmitters. In the first 2D cubic-spline (2D-CS) representation, the gain functions must be extracted along each axis sequentially. The gain values at the knots should be calculated using an alternate basis, typically 2D polynomials. Secondly, in the new 2D least-square cubic-spline (2D-LSCS) approach, a 2D cubic-spline basis is introduced such that the basis weights can be extracted directly from measured data using the least square method. Inside the FPGA, the 2D basis functions are calculated from 1D basis functions to reduce the signal-processing resource usage in the real-time implementation. Two different test scenarios involving 3 carrier WCDMA and long term evolution (LTE) signals which are 310 MHz apart are considered. The experimental results show that the in-band intermodulation distortion products are reduced by 20 dBc, with an ACPR by less than -50 dBc and NMSE by less than -40dB for a 10 W dual-band PA. The 2D-LSCS basis improves performance by upto 3 dB in both ACPR and NMSE when compared to the conventional 2D polynomial model and 2D-CS approaches. At the same time, it uses reduced FPGA resources and features a faster extraction process.
This is dedicated to my wife Navya and my Family.
ACKNOWLEDGMENTS

I would like to thank my advisor Prof Roblin for his belief in me and his guidance in both academic and personal. His never ending learning process and patience in explaining things motivated me to perform my research with dedication and sincerity. He introduced me to the world that used digital signal processing for RF applications specifically for linearizing the power amplifier. He taught me to come up with quality writing skills for the technical papers. I would like to thank Prof Joanne E.Degroat and Prof Furrukh Khan for being part of my qualifiers, candidacy, and defense committee. Their in depth knowledge of FPGA implementation and valuable suggestions has improved the designs during the optimization stage.

The most important person who was involved during the crucial time in my PhD is Dr. Christophe Quindroit. He drives me to think and tackle the problem in the most efficient way. The long conversations I had with him has improved my knowledge in the field of power amplifiers and digital predistortion. I would also like to thank my lab members Dr. Meenakshi Rawat, Haedong Zang, Youngseo Ko, Yiqiao Lin, Robert Pond and Jiwoo Kim who were involved with me at some point of time during my journey.

I would like to thank Analog Devices Inc., Wilmington, MA and NXP Semiconductors, Smithfield, RI, for donating the Mixed Signal Digital Pre-distortion System Boards (MSDPD) and the PAs respectively which are used in this research. I would
also like to thank Altera Corporation-Wireless Systems Solutions Group for their financial and technical support in this research and the donation of the Stratix IV FPGA. This research was also supported in part by the National Science Foundation under grant ECS 1129013.

I would like to thank my wife Navya for her support in proof reading all my papers. I would also thank my dad (Kaleswara Rao), mom(Sai Syamala) and my brother(Yaswanth) for their love and support. I thank my dad for believing in me and supporting financially during early stages when there was limited funding from school. I thank my brother Sainath Nutakki and Meenakshi Bandreddi for motivating to start my Ph.D. program.
VITA

June 12, 1984 ...............................Born - Andhra Pradesh, India

Jun, 2005 .....................................B.Tech. Electronics and Communications Engineering, Acharya Nagarjuna University

Apr, 2006 - Apr, 2007 ......................Associate Software Engineer, Infor Global Solutions

Dec, 2008 .......................................M.S. Electrical Engineering, University of Michigan

Mar, 2011-May, 2012 .........................Graduate Teaching Associate, The Ohio State University.

Jan, 2012-Dec, 2013 .........................Graduate Research Associate, The Ohio State University.

Jan, 2014-May, 2014 .........................Graduate Teaching Associate, The Ohio State University.

May, 2014 .......................................M.S. Electrical Engineering, The Ohio State University

May, 2014 - Present .........................Digital RF Communications Engineer, GatesAir

PUBLICATIONS

Research Publications


**FIELDS OF STUDY**

Major Field: Electrical and Computer Engineering

Studies in Non-Linear RF Lab: Prof Patrick Roblin
**TABLE OF CONTENTS**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Dedication</td>
<td>iv</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>v</td>
</tr>
<tr>
<td>Vita</td>
<td>vii</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xii</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xiii</td>
</tr>
<tr>
<td>Chapters:</td>
<td></td>
</tr>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Wireless Communication</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Digital Front End</td>
<td>2</td>
</tr>
<tr>
<td>1.2.1 DDC and DUC</td>
<td>4</td>
</tr>
<tr>
<td>1.2.2 ADC and DAC</td>
<td>6</td>
</tr>
<tr>
<td>1.3 Power Amplifier</td>
<td>7</td>
</tr>
<tr>
<td>1.3.1 Memory effects</td>
<td>11</td>
</tr>
<tr>
<td>1.4 Emergence of Multiband PA and DPD</td>
<td>13</td>
</tr>
<tr>
<td>1.5 Digital Predistortion</td>
<td>15</td>
</tr>
<tr>
<td>1.5.1 Frequency Selective Technique</td>
<td>16</td>
</tr>
<tr>
<td>1.5.2 Widely Separated Carriers</td>
<td>17</td>
</tr>
<tr>
<td>1.5.3 Time Alignment</td>
<td>19</td>
</tr>
<tr>
<td>1.5.4 Figure of Merits</td>
<td>21</td>
</tr>
<tr>
<td>2. Robust PA Modeling [1]</td>
<td>24</td>
</tr>
</tbody>
</table>
3. Quasi Exact Inverse ............................................ 30
   3.1 Single Band System [2] ..................................... 31
      3.1.1 Conventional Indirect Learning ...................... 31
      3.1.2 Quasi Exact PA Inverse ................................ 33
      3.1.3 Measurement Results .................................. 36
   3.2 Dual band System [3] ...................................... 39
      3.2.1 Indirect Learning ..................................... 40
      3.2.2 2D Quasi Exact Inverse .............................. 41
      3.2.3 Measurement Results .................................. 44

4. Cubic Splines .................................................. 46
   4.1 Single-Band System ....................................... 47
      4.1.1 Least Square Cubic Spline .......................... 48
      4.1.2 Measurement Results .................................. 50
   4.2 Dual-Band System .......................................... 50
      4.2.1 2D Conventional Cubic Spline (2D-CS) [4] ............ 52
      4.2.2 2D Least Square Cubic Spline (2D-LSCS) ............. 55
      4.2.3 Measurement Results .................................. 64

5. Design Methodology ............................................ 73

6. Conclusion ..................................................... 80

7. Future Work ................................................... 83
   7.1 Full Transmitter .......................................... 83
      7.1.1 Crest Factor Reduction ............................... 83
      7.1.2 Digital Up Converter (DUC) .......................... 84
   7.2 Full Receiver ............................................... 84
      7.2.1 Digital Down Converter (DDC) ....................... 84
      7.2.2 Time-alignment ....................................... 84
      7.2.3 Matrix Inversion ..................................... 85
   7.3 New Algorithms ............................................. 85
      7.3.1 Dual Feedback Architecture .......................... 85
      7.3.2 Reduced Complexity for Multiband DPD ............... 87
      7.3.3 Adaptation ............................................ 88

Bibliography ..................................................... 89
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Comparison of NMSE and ACPR for Single Band System</td>
</tr>
<tr>
<td>3.2</td>
<td>Comparison of NMSE and ACPR for -50 dBc Noise Floor for Dual-Band system</td>
</tr>
<tr>
<td>4.1</td>
<td>Comparison of NMSE and ACPR for Single Band System</td>
</tr>
<tr>
<td>4.2</td>
<td>NMSE/ACPR for Dual-Bands with/without DPD</td>
</tr>
<tr>
<td>4.3</td>
<td>Total number of coefficients for each model</td>
</tr>
<tr>
<td>4.4</td>
<td>Comparison of the DPD extraction time and real-time DSP resource utilization</td>
</tr>
<tr>
<td>4.5</td>
<td>FPGA resource utilization</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Block Diagram of SDR system</td>
<td>3</td>
</tr>
<tr>
<td>1.2</td>
<td>Digital Down Conversion</td>
<td>5</td>
</tr>
<tr>
<td>1.3</td>
<td>Digital Up Conversion</td>
<td>6</td>
</tr>
<tr>
<td>1.4</td>
<td>$P_{out}$ and PAE vs $P_{in}$</td>
<td>9</td>
</tr>
<tr>
<td>1.5</td>
<td>IMD Generation: Two Tone</td>
<td>10</td>
</tr>
<tr>
<td>1.6</td>
<td>IMD Generation: Dual band</td>
<td>11</td>
</tr>
<tr>
<td>1.7</td>
<td>Memory model of a Power Amplifier</td>
<td>12</td>
</tr>
<tr>
<td>1.8</td>
<td>The composite RF signal $\Re {x_1 e^{i\omega_1 t} + x_2 e^{i\omega_2 t}}$ (blue) for two complex modulated signals $x_1(t)$ and $x_2(t)$ are plotted together with the RF envelope $</td>
<td>x_1 e^{i\omega_1 t} + x_2 e^{i\omega_2 t}</td>
</tr>
<tr>
<td>1.9</td>
<td>General DPD System</td>
<td>15</td>
</tr>
<tr>
<td>1.10</td>
<td>Cascade System (DPD+PA) for Linearizing PA</td>
<td>16</td>
</tr>
<tr>
<td>1.11</td>
<td>Comparison between Single and Frequency Selective DPD architecture for Dual Band System</td>
<td>18</td>
</tr>
<tr>
<td>2.1</td>
<td>Distribution of the envelopes at PA input and output for two independent LTE signals at LSB and USB respectively</td>
<td>26</td>
</tr>
</tbody>
</table>
2.2 Distribution of the envelopes at the PA input and output for a specially synthesized pair of multisines at LSB and USB respectively. Normalized input envelopes with an average power $|x_1|^2 + |x_2|^2$ in the same power range are plotted using the same color group on the left graph. An increment of 0.2 in normalized envelope is used. The corresponding normalized PA output envelopes are shown using the same color on (b) graph. The input and output envelopes are normalized relative to the peak envelope in each band. .................................................. 27

2.3 (a) 3D plot of envelope of $|y_1|$ and (b) $|y_2|$ at the PA output for memoryless PA model. Measured data (red dots), the extrapolated data (black circles) and the cubic-spline fit (lines) are compared ....... 28

3.1 Different Learning Schemes ........................................ 31
3.2 Block Diagram of Single Band Test Bench ................. 32
3.3 Testbed Setup ...................................................... 35
3.4 Measurement Results: Spectrum ................................ 36
3.5 Measurement Results: In/Out, AM/AM and AM/PM curves of the PA .......................................................... 37
3.6 Block Diagram of Dual Band Test Bench ..................... 39
3.7 In-Direct Learning .................................................... 40
3.8 Quasi Exact Inverse ............................................... 41
3.9 Test Bench Setup .................................................... 44
3.10 Measurement Results: Spectrum ............................... 45
4.1 Graphical representation of the gains $G_p^{(m,u,v)}(|x_1|^2, |x_2|^2)$ as a function of $|x_1|^2$ and $|x_2|^2$ for all memory delay indices $m$ ......... 47
4.2 1D Basis Function at each knots ............................... 49
4.3 Measurement Results: Spectrum of 3c-WCDMA ............ 51
4.4 2D-CS implementation with memory for each band $i$ .... 56
4.5 2D basis functions \( \phi_{ij} \) of LSCS for each knot along \( |x_1|^2 \) and \( |x_2|^2 \). 60

4.6 2D-LSCS basis computation ........................................... 62

4.7 Tensor Product of \( \phi_i \) and \( \phi_j \) ........................................... 63

4.8 FPGA and MSDPD clock Architecture .............................. 65

4.9 Comparison of the LSB and USB spectral performance of the 2D-LSCS, 2D-CS and 2D conventional polynomial models for the 2 test cases. 66

4.10 Comparison of the LSB and USB spectral performance of the 2D-LSCS and 2D conventional polynomial models for the 2 test cases when they are implemented in the FPGA testbed using the hybrid real-time architecture. ........................................... 67

5.1 Altera Wireless RF Framework [5] .................................. 75

5.2 Qsys System ................................................................. 77

5.3 FLow of Design Methodology ........................................ 78

5.4 DPD design using DSP Builder ....................................... 79

5.5 DPD design using Quartus ............................................ 79

7.1 Dual Feedback Architecture using QEI Predistoter ............... 86

7.2 Simulation Results: Single and Dual DPD ............................ 87
CHAPTER 1

Introduction

1.1 Wireless Communication

In the last few decades the amount of data transferred is significantly increased due
to generating high resolution photographs and videos etc. These large amounts of data
are transferred from one location to another using wireless channels like wireless local
area networks (WLAN). Due to this increase in data traffic the frequency spectrum
is congested with more number of channels. Previously frequency modulation tech-
niques were used because of its advantages like susceptible to noise, interference etc.
But due to its wider bandwidth the spectral efficiency is very poor. To make possible
this global wide-bandwidth wireless communication and networking while handling
the increasing number of users, new wireless communication standards based on high
bandwidth efficiency protocols such as Orthogonal Frequency Division Multiplexing
(OFDM) and Code Division Multiple Access (CDMA) have been developed to meet
the user capacity requirements. Multicarrier modulation technique like OFDM also
provide advantages over wideband single carrier systems like:

i) Improving the spectral efficiency by allowing overlapping of the sub-carriers
ii) OFDM is more resistant to frequency selective fading by dividing the wide band channel into narrow band flat fading subchannels.

iii) Inter symbol interference (ISI) is eliminated by using cyclic prefix.

iv) Channel equalization becomes simpler.

However these new communication standards have placed very challenging demands on the RF front-end specifications in terms of power efficiency for both handheld devices and base stations. The inherent characteristic of OFMD signal is that it has very high peak \( P_{\text{max}} \) to average \( P_{\text{avg}} \) power ratios (PAPR), typically on the order of 10 dB. The PAPR of the signal is defined as:

\[
PAPR(dB) = 10 \cdot \log_{10} \frac{P_{\text{max}}}{P_{\text{avg}}} \tag{1.1}
\]

Thus for a PA with an average power of 80W should be also be able to amplify linearly outburst, with 800W instantaneous peak power. Special RF amplifiers are then needed to amplify such signals with large PAPR while providing high average power efficiency [6]. These complex modulation techniques thus feature a non-constant envelop which imposes strict linearity requirements on the PA.

1.2 Digital Front End

A superhetrodyne architecture including a digital radio transceiver is illustrated in Fig. 1.1. In a superhetrodyne transmitter the base band signal is first converted to a low-IF frequency and then it is modulated onto a high frequency carrier. As shown in Fig. 1.1 the hardware parts of the RF, IF and baseband sections are all controlled
and programmed using a software. This kind of architecture is referred to as software defined radio (SDR).

The RF front-end is the analog section which includes (not shown in the Figure) LNA, mixer, filters, voltage controlled oscillator (VCO), PA, and antenna. This section is responsible for conversion between RF and IF signals. The advanced RF front-ends also have some degree of controllability using software, e.g. frequency tuning, I/Q imbalance tuning, gain control etc.

All the other parts of the architecture comprise of digital processing components. The list of operations performed at the IF domain includes:

i) Sampling and separation of IF carriers.

ii) DDC and DUC (digital down converter and digital up converter)
   a) Digital Synthesizer or Digital numerically controlled oscillator (NCO)
   b) Digital mixer
c) I/Q modulation/demodulation

d) Multirate decimation filtering

An FPGA is generally used to perform these operations because of the high speed sampling and digital conversion capabilities of the current FPGA modules.

Finally the back-end of the SDR architecture defines the baseband process which performs:

i) Symbol timing recovery

ii) Equalization

iii) Modulation and Demodulation

iv) Encoding and Decoding

v) Channel coding

In a conventional method, an analog signal is passed through individual hardware components performing a specific task. The SDR architecture uses FPGA and DSP to generate transmitted signal and detect receive signal. The goal of the SDR system is to implement a radio system which is flexible, versatile and multi standard. The goal is achieved by carrying the necessary operations digitally using a software.

1.2.1 DDC and DUC

In the transmitter, a DUC in DSP is used to convert from zero intermediate frequency zero-IF to low-IF. Whereas a DDC is present in receivers to convert a digital signal low-IF to zero-IF. Complex numbers are used to represent wireless signals like
WCDMA, LTE etc. The real components are defined as In-phase (I) component and quadrature phase (Q) component.

The main components of DDC as shown in Fig. 1.2:

1. Digital Mixer

2. NCO

3. Sample rate conversion

If the conversion rate is an integer, the sample rate conversion includes filtering the image signal and sample rate conversion using anti-aliasing filter and down-sampler. But when the conversion rate is not an integer the sample rate conversion is performed by first up sampling, filtering using anti aliasing filter and then perform the down sampling.

The block diagram of the DUC also contains similar blocks as DDC but in the opposite direction as shown in Fig. 1.3
1.2.2 ADC and DAC

The interface between analog and digital domain includes mixed-signal block which converts digital base band signal into high resolution analog signal (DAC) at the transmitter and high resolution analog signal to digital base band signal (ADC) at the receiver. The different parameters used to define these blocks are sampling frequency, resolution, conversion time etc.

The frequency at which sampling is performed is defined by the sampling frequency. As defined by Nyquist-Shannon theorem, this frequency should be at least twice the highest frequency of the bandlimited signal for error free recovery of the signal. The maximum sampling frequency of the DAC that is being used here is 1.2 GHz designed by Analog Devices (ADi). The resolution is defined by the number of bits that are used for representation. The more the number of bits the lesser is the quantization noise. The number of bits for DAC used is 16 bits and 12 for ADC.
The sub-sampling principle of ADC is used in the receiver architecture to reduce the sampling frequency required in the feedback path. The received low-IF signal is at 184.32MHz which falls exactly at the center of the second Nyquist zone of the ADC operating at 250 MSPS.

1.3 Power Amplifier

Power amplifier (PA) is the essential part of any communication system to amplify the signal for long distance transmission. There is trade-off in the design of the PA with respect to linearity and efficiency. Advanced design techniques are used to improve the efficiency of the PA using architectures like Doherty and envelope tracking [7]. The efficiency of the PA is increased at the expense of non-linearity. A spectral regrowth within the bandwidth and also outside the signal bandwidth of the amplified signal are generated due to this non-linearity. The in-band spectral regrowth decreases the bit error rate (BER) at the receiver and out of band spectral components causes distortion or interference with adjacent channels. The multicarrier and spectral efficient modulated signals have a very high peak average power ration (PAPR) i.e. large fluctuation in their signal envelopes, thus operating the amplifier in the non-linear region creates the unwanted intermodulation products (IMDs). There are several solutions for this problem. We can use a more linear but inefficient amplifier thus increasing the heating problems and power consumption. These devices cannot be used for portable devices because it drains the battery very fast. We can change the modulation scheme to be more robust but it would be less spectral efficient. We can also increase the frequency spacing between the channels which also reduce the spectral efficiency. And moreover signal spacing and modulation methods depends
on the communication standard used. Another simple technique to avoid operating the amplifier in non-linear region is to operate at back-off from the saturation region. The efficiency of the power amplifier will be greatly reduced due to the back-off. As a result, most of the DC power is dissipated in the form of heat. The power added efficiency (PAE) of an amplifier is given in (1.2):

\[
PAE(\%) = \frac{P(RF_{out} - P(RF_{in}))}{P_{dc}}
\]  

(1.2)

PAE defines the measure of the power conversion efficiency of the power amplifier. Ideally, all the supplied power to the amplifier will be converted into output power. PAE can be interpreted as the efficiency of the network to convert the input DC power into the amount of output RF power that is left over after the direct contribution from the input power is removed. With linearization techniques the PA can be operated more close to the saturation region with high linearity and efficiency. DPD is most widely used linearization technique for its robustness, less complexity and ease of implementation. In modern communications and radar applications PAs are the most DC power consuming block among all the subsystems. Considering the number of base stations that are present all over the world, if the DC power is used efficiently, there is drastic reduction in the electricity and cooling costs for the service providers.

The red curve in Fig. 1.4 represents the PAE of the PA. As it can be observed that the PAE is very small for lower power and it increases as the PA is operated close to the saturation region. But non-linearity of PA will come into the effect thereby distorting the signal.

The complex modulation techniques like quadrature amplitude modulation (QAM) which are spectral efficient uses both the amplitude and phase to carry the message
signal. Since the information is stored in the amplitude for these techniques, they are very sensitive to amplitude fluctuations because of the non-linear amplification of the PA. The energy efficiency is a very important factor for mobile devices for better usage of the battery. Moreover since PA is the most power consuming component of the transceiver chain, choice has to be made between linearity and power consumption.

When two tones are transmitted through the non-linear PA, the IMDs are generated at the output at a distance of frequency spacing between the 2 tones as shown in Fig. 1.5. This can be generalized to the amplification of broad band signal where we have an infinite number of tones in the bandwidth. Now these IMDs fall in the bandwidth and also on the adjacent channels, this effect is called spectral spreading or regrowth. When a single band is passed through a non-linear PA there is spectral

Figure 1.4: $P_{out}$ and PAE vs $P_{in}$
regrowth within the band thereby decreasing the ACPR of the signal. Hence the signal will not pass the spectral mask created by Federal Communications Commission (FCC) standards. The non-linear behavior is more prominent in dual band systems in which two modulated signals that are separated in carrier frequencies by $\Delta \omega$ are transmitted simultaneously. When dealing with dual-band transmitters the IMDs can be categorized into three major groups as shown in Fig. 1.6

1. In-band intermodulation: This distortion consists of products around the carrier frequency that are solely due to the intermodulation between the signal elements within each band which is similar to the single band transmitters.

2. Inter/out of band intermodulation: The IMD products that are between the two signals in both the frequency bands and they are located $\Delta \omega$ away from the lower and upper carrier frequency.

3. Cross modulation: These products appear within the same frequency range as in in-band intermodulation, however the distortion is the result of intermodulation between the signals in both the frequency bands.
1.3.1 Memory effects

A behavioral model for PA can be classified as:

1. Memoryless: Modeling for static AM/AM and AM/PM curves of the PA i.e. the output depends on the instantaneous input.

2. Memory: Memory effect within a PA can be modeled by applying filtering to a memoryless nonlinear block before and after as shown in Fig. 1.7. Nonlinear memory can be introduced because of the unwanted gain modulation which is a function of present and past envelops of the input. Due to the presence of nonlinear block and the fact that it doesn’t commute with nonlinearities, they need to be considered separately. Simplified memory models uses only one of the filter for moderate non-linear and memory PAs.

The DPD performance is improved with inclusion of memory effects (also referred as time-selectivity [8]) specifically for broadband PA excited by wide band signals. With demanding high-data rate and broadband services in wireless communications, RF PA are exhibiting frequency or history dependent behavior. This behavior has short and long time constant when compared to the RF carrier signal or its slowly
Figure 1.7: Memory model of a Power Amplifier

Short term memory effects are caused by the band pass characteristics of the PA input and output matching networks, and sometimes on the low pass characteristics of the active device. The long term effects are due to the thermal time constant of the device and some of the components of the biasing circuits. The effect of memory effect is visible on the memoryless curve as dispersion and unequal spectral regrowth in the IMDs. To effective linearize the wideband signals, these memory effects should be included in the DPD model [9].

Different types of memory effects are classified as:

1. Fast memory effects which are typically associated with the rapid response of the PA to the signal given its wide modulation bandwidth (MHz)

2. Slow memory effects such as traps and self-heating, which are associated with the slow response of the PA system to the long term variation of the envelope of the applied modulated signal.
Slow memory effects can usually be dealt with using adaptation since they involve long time constants, provided a fast enough adaptation is developed. However, fast memory effects excited by the modulation bandwidth (typically 5 to 100 MHz) around each carrier are much too fast to be remediated by using adaptation.

Many band-limited techniques have been proposed for addressing the faster memory effects associated with each band. One of the most convenient and efficient technique to implement is that of memory polynomials [10] [11].

1.4 Emergence of Multiband PA and DPD

With the explosive growth of the smart phone and tablet markets, wide bandwidth voice and data communication have become ubiquitous. Users expect to use their wireless portable phone/computing devices at any place and at any time. To handle the increasing number of users, the aggregated bandwidth of multicarrier base-station signals keeps increasing. For example, systems with 100 MHz useful bandwidth requiring 500 MHz of linearization bandwidth are being developed. With the current advances in the design of broadband PAs operating from several MHz to several GHz, new transmitter architectures are being deployed using a single PA accommodating two concurrent bands. The increased bandwidth places technical challenges on the DPD performances as it must handle the PA memory effects for wider bandwidths. Furthermore, due to market dynamics, cellular phone service providers may end up operating base-stations in different non-contiguous frequency bands. Tremendous saving can therefore be achieved when new generations of basestations are deployed using single multiband RF power amplifier. This has motivated significant research in the field of multiband linearization.
As the carrier separation between the dual-band increases, the effective modulation bandwidth \( f_2 - f_1 \) becomes very large. Indeed, when two modulated carriers \( x_1 \) and \( x_2 \) are combined, the composite RF signal will be beating at the frequency \( f_2 - f_1 \). That is, as is illustrated in Fig. 1.8, the envelope (red line) of the composite RF signal (blue line) will rapidly oscillate at the frequency \( f_2 - f_1 \). Note that the envelope of the composite RF signal oscillates between its peak value (green line) and its minimum value (yellow line) when the two RF signals from the two bands adds constructively (in phase) and destructively (out of phase) respectively.

Figure 1.8: The composite RF signal \( \Re \{ x_1 e^{i \omega_1 t} + x_2 e^{i \omega_2 t} \} \) (blue) for two complex modulated signals \( x_1(t) \) and \( x_2(t) \) are plotted together with the RF envelope \( |x_1 e^{i \omega_1 t} + x_2 e^{i \omega_2 t}| \) (red line) the peak envelope \( |x_1| + |x_2| \) (green line) the average envelope \( \sqrt{|x_1|^2 + |x_2|^2} \) (purple line) and the minimum envelope \( ||x_1| - |x_2|| \) (yellow line)
1.5 Digital Predistortion

A DPD processes the base-band signal to create an expanded non-linearity that is complimentary to the compressing characteristics of the PA. Ideally, the cascade of the predistorter and PA should amplify a signal with a constant gain. The PA can be utilized near the saturation region but still maintain a proper linearity using DPD. The general DPD architecture is shown in the Fig. 1.9. The predistortion block in the forward path is the block where the extracted coefficients are applied to the model to predistort the signal. After the signal is predistorted it is converted into analog signal and upconverted to RF frequency to be transmitted through the PA. There is an observation path which observes the transmitted signal and downconvert to base-band signal in order to compare with the input signal. The predistorter construction extracts the coefficients by using the indirect/direct learning technique as discussed in Chapter 3 and 4.

The non-linearity is characterized by amplitude modulation of input to the gain (AM/AM) and also the phase modulation of the gain with the amplitude modulation of the input (AM/PM). The characteristic curves of the DPD, PA and their combined
The different techniques that are used to lighten the complexity of a multiband DPD system are explained in the subsequent section.

1.5.1 Frequency Selective Technique

There are difficulties in using conventional DPD technique for compensating multiband system. Using a single DPD model to compensate for all the distortion requires capturing the whole spectrum at the output of the PA. The bottleneck of this approach is the sampling rate limitations and expensive ADC/DAC for high sampling frequencies. Using two independent conventional DPDs are also not sufficient to
compensate the distortion because the cross modulation products are not taken into account. A new architecture which uses a frequency selective technique can be used in order to relax the requirements on the sampling frequencies of ADC/DAC.

A 2D-DPD architecture can be developed to compensate for the distortion and nonlinearities of dual-band transmitters. The 2 processing cells are responsible for the compensation of the non-linearity associated with that particular frequency band. Even though this architecture has DPD block for each band, the cross modulation terms from the other band are still considered as used in (1.3). Using this technique the sampling rates of ADC/DAC are greatly reduced [12]. The output of the memory polynomial model is given by (1.3):

\[
y_1(n) = \sum_{m=0}^{M-1} \sum_{k=0}^{N-1} \sum_{j=0}^{k} c_{jkm}^{(1)} |x_1(n - m)|^2 |x_2(n - m)|^{2(k-j)} x_1(n - m)
\]

\[
y_2(n) = \sum_{m=0}^{M-1} \sum_{k=0}^{N-1} \sum_{j=0}^{k} c_{jkm}^{(2)} |x_2(n - m)|^2 |x_1(n - m)|^{2(k-j)} x_2(n - m)
\]

(1.3)

Where \( y_1(n) \) and \( y_2(n) \) are the sampled received signals, \( x_1(n) \) and \( x_2(n) \) are the sampled input signals, \( c_{jkm}^{(1)} \) and \( c_{jkm}^{(2)} \) are the coefficients of the memory polynomial, \( |x_1(n)| \) and \( |x_2(n)| \) are the absolute value of complex envelops for lower band and upper band respectively.

An approximation for the coefficients of the inverse model are determined either by using indirect learning or direct learning techniques. The extracted coefficients are updated in the DPD blocks to predistort the signal before sending it to the PA.

### 1.5.2 Widely Separated Carriers

When a band limited signal is passed through a nonlinear PA, IMDs are generated as shown in the Fig. 1.6. The bandwidth of the IMD waveforms are determined by
the polynomial order of the nonlinearity e.g IMDs associated with a non-linearity of order $P$ will occupy $P$ times the bandwidth of the linear signal. However much of the IMD power is concentrated near the channel containing the linear bandwidth. The RF transmitter may transmit several carrier signals occupying different channels. The IMDs can be originated from the intermodulation within the individual carriers (inband) as well as intermodulation between carriers (interband). The spread of distortion is large for interband than the inband distortion. The inband distortion is going to affect the BER and error vector magnitude (EVM) of the received signal whereas the interband distortion creates interference in the neighboring channels. In
order to compensate for these IMDs the DPD has to create the IMDs which are of same magnitude but opposite in phase to be canceled. So to sample the entire spectrum, the DPD module has to sample at very high rate i.e. several times larger than the Nyquist rate associated with the band limited linear signal to obtain cancellation over the entire IMD bandwidth. The higher sampling rate increases the cost of components such as the ADC and DAC resulting in a very expensive transmitter. The complexity of the DPD system is reduced by increasing the frequency spacing between the two bands so that the interband distortion products fall far from the message signal. The products can be easily removed using a transmit filter which is placed after the PA, and the skirt selectivity of this filter doesn’t have to be very sharp [13]. Since the filter uses a low Q-factor design, it is very cost effective thereby reducing the cost of the system.

1.5.3 Time Alignment

To measure the complex gain compression curve of PA using the input and output waveforms, they need to be perfectly aligned for accurate prediction. If there is error in the delay calculation, it introduces significant dispersion in the AM-AM and AM-PM curves which is wrongly interpreted as memory effect. This misalignment also degrades the prediction capability of the predistorter which is compensating for the non-linear characteristics of the PA.

A maximum cross correlation between the input and output is used in [14] for estimating the delay. A two step method is proposed using coarse delay tuning and fine tuning delay. A Lagrange interpolation is used at the input and output signals for
fine tuning the delay. For estimating high precision delay, large interpolation factor is required which greatly increases the computational resources.

To alleviate this problem the characterization process proposed in [15] and [16] is used to estimate the delay introduced by the DUT in single step. Most of the previous methods uses time domain alignment compensation but here a frequency domain alignment is used for better accuracy [16].

Consider \( x(t) \) and \( y(t) \) are the baseband transmitted and received signal that needs to be time-aligned for prediction of PA or PD.

\[
y(t) = g \cdot x(t - \tau) \cdot e^{j\varphi_0} \tag{1.4}
\]

where \( \tau \) and \( g \) are time delay and gain of the path respectively, and \( \varphi_0 \) is the phase difference between the local oscillators of transmitter and receiver.

Let us consider \( X(f) \) and \( Y(f) \) to be the Fourier transform (FT) of the baseband signals \( x(t) \) and \( y(t) \). Consequently, the expression can be written as:

\[
Y(f) = g \cdot X(f) e^{j(\varphi_0 - 2\pi f \tau)} \tag{1.5}
\]

The Fourier transform of cross correlation of signals is given as:

\[
(x(t) \star y(t)) \xrightarrow{FT} X(f) \cdot Y^*(f)
\]

\[
(x(t) \star y(t)) \xrightarrow{FT} g \cdot X(f) \cdot X^*(f) \cdot e^{-j(\varphi_0 - 2\pi f \tau)} \tag{1.6}
\]

\[
(x(t) \star y(t)) \xrightarrow{FT} g \cdot |X(f)|^2 \cdot e^{-j(\varphi_0 - 2\pi f \tau)}
\]

The average delay \( \tau \) in the signal bandwidth and the phase rotation \( \varphi_0 \) can be computed from the phase of (1.6) as \(-(\varphi_0 - 2\pi f \tau)\). The delay is compensated by
multiplying the output $Y(f)$ with $e^{(\varphi_0 - 2\pi f\tau)}$. The output is aligned with the input once the delay is compensated.

1.5.4 Figure of Merits

Figure of merits are used to quantify and compare the performance of different algorithms used for PA modeling and DPD. Since predistortion is also based on the principle of modeling the PA, all the metrics that are used to evaluate any PA modeling technique can be used for predistortion models also. Different figure of merits used in this work are given below:

**NMSE**

A metric that is used to evaluate any predistorter model is normalized mean square error (NMSE). An NMSE defines how well a modeled data is calculated when compared to the actual measure data. The smaller the NMSE value the better the model.

\[
NMSE = \frac{\sum_{n=0}^{N-1} |y_{meas}(n) - y_{fit}(n)|^2}{\sum_{n=0}^{N-1} |y_{meas}|^2} \quad (1.7)
\]

Where $y_{meas}(n)$ is the actual measure data that is received while $y_{fit}(n)$ is the fitted data using the polynomial model.

**ACPR**

Another metric to be considered is ACPR which is defined as part of signal that falls in adjacent signal band in relation to the signal power on the signal bandwidth. The ACPR also confirms whether the signal meets the FCC mask for a particular
standard. Each standard has a specification for ACPR(db). A typical value of ACPR is around -45 dbc.

\[
ACPR(dB) = 10 \log_{10} \left( \frac{P_c}{P_{adj}} \right) \tag{1.8}
\]

Where \( P_c \) is the channel power and \( P_{adj} \) is the adjacent or out of band power beyond the bandwidth.

In Chapter 2, a 2D multisine excitation signal is proposed to assist with the system identification of 2 band PAs. Two training multisines \( x_1 \) and \( x_2 \) with average power and peak to average power (10dB) matching the targeted LTE communication signals are used to modulate the lower and upper bands. The multisines are also selected to provide individually a good 1D mapping of the \( x_1 \) and \( x_2 \) state space. Some important behaviors of the PA excited with concurrent dual-band is explained in this chapter.

In Chapter 3, a new single band DPD algorithm for PA model with an arbitrary number of memory delays is experimentally investigated. This DPD algorithm is based on the quasi-exact inverse (QEI) of the PA model which achieves typically less than -84 dB NMSE in simulations when applied to the PA model itself [2]. The experimental verification of the model is performed on a testbench setup which closely resembles to a real base station. Most of the DPDs in the current literature depend on vector signal generator (VSG) and vector signal analyzer (VSA) which exhibit very high performance but are not cost effective. The advantage of using a real system is that the non-idealities of the system can be accounted for and mitigated by the DPD algorithm. The testbench consists of a FPGA which predistorts the signal and send it to analog device’s mixed signal DPD (ADI MSDPD) board for upconversion to an RF signal. The RF signal is sent to PA for amplification and returned to MSDPD.
board for downconversion. The low IF signal is stored on to the DDR3 memory of the FPGA for further processing. The aim of this work is to reproduce a realistic base-station operation where the communication signals sent are played only once and the linearization must be performed in real-time. This is also extended to accommodate concurrent dual-band DPD verification using 2D QEI algorithm.

In Chapter 4, the advantages of piece-wise polynomials over regular polynomials is presented. Two new 2D C-spline models are developed which models the gain of the amplifier with memory delays in order to take into account the memory effects in the dual-band PA. A 2D-LUT model is designed which can be used to store the spline coefficients instead of the gain values to reduce the memory usage. Theoretical equations to create the 2D C-spline for modeling the gain of the PA and DPD are derived. A hybrid architecture is also used as an alternative implementation, where the 1D basis functions created by cubic spline are stored in the memory and the 2D basis function is created in real-time. The experimental measurement results are then reported to compare the performance of these two methods with a conventional polynomial model. Finally the results and resource utilization obtained for the new proposed 2D C-spline linearization are summarized.

In Chapter 5, the design methodology and different tools that are being followed and used in this work to reduce the development life cycle is described. The thesis is concluded in Chapter 6 by stating some observations and also comparing different algorithms and architectures performance. Chapter 7 lists some of the topics and ideas that can be extended and developed in future.
CHAPTER 2

Robust PA Modeling [1]

The general frequency and time selective theories which are used for modeling the PA as well as performing linearization using DPD is presented in Chapter 1. When modeling the PA or extracting the DPD algorithm, the gain functions $G_{1,m}(|x_1|^2, |x_2|^2)$ and $G_{2,m}(|x_1|^2, |x_2|^2)$, for the two bands must be extracted for each delay $m$. Various techniques are possible for representing these gain functions. The most common approach relies on polynomials of the envelopes squared $|x_1|^2$ and $|x_2|^2$ [17] or the envelope $|x_1|$ and $|x_2|$ for improved non-analytical fitting. Nonetheless, polynomial expansions are profitably used for the gain functions due to their simplicity and high performance. Polynomials work particularly well for the DPD linearization. When the gain function for the PA saturates, then the DPD stage diverges at large inputs. Polynomials which are prone to divergence are therefore very comfortable with the DPD gain behavior.

Still, there is room for improvement with polynomials with a careful choice of the polynomial basis selected. Orthogonal polynomials have been shown to be useful in single band DPD linearization [18]. Similarly, it is possible to extend these results to two bands or more [19] [20]. Indeed, owing to the statistical independence of the various bands making up the composite signal, the orthogonal polynomials developed
for a single band can be directly applied without any modification. It is sufficient
to generate a tensor product to apply them in the multiband case. When using the
orthogonal polynomials, the numerical condition number of the matrix used in the
linear least square matrix solution is found to be greatly decreased. Still a question
arises on the way orthogonal polynomial benefits the linearization, given that or-
thogonal polynomials are obtained using a linear superposition of the various power
terms. For ideal theoretical DSP systems with floating or wider fixed point accuracy,
no effective improvement is to be expected. However, when using a reduced num-
ber of bits as in the case of practical DSP systems, due to its increased efficiency,
the orthogonal representation provides substantial improvements when measured in
terms of NMSE and reduced number of iterations [19] [20]. Further improvements
can be obtained using an iterative method developed to prune the 2D DPD model to
reduce the needed number of coefficients [21]. An artificial neural networks (ANN)
have recently been reported for multiband DPD [22] and have been demonstrated to
deliver high linearization performance like in the single band DPD [23]. ANNs do
provide continuous derivatives of infinite orders and naturally exhibits graceful degra-
dation, however, the ANN learning is usually time consuming. A dual band forward
twin nonlinear two-box (2D-FTNTB) model using orthogonal polynomial is proposed
in [24] which improves the 2D Hammerstein model while reducing the complexity of
the 2D-DPD model.

An LUT, cubic-spline or B-spline can also be used as an alternative technique
for representing the gain functions $G_{1,m}(|x_1|^2, |x_2|^2)$ and $G_{2,m}(|x_1|^2, |x_2|^2)$, [4] [25].
Spline provides an improvement in NMSE for the power amplifier model and also for
linearization. Using the polynomial representation, it is possible to directly extract
the spline coefficients using a linear least square inversion of a matrix as discussed in Chapter 4. However, in 2D, due to the scarcity of the data at high powers when the output powers of both bands are high at the same time, the conditioning of the B-spline matrix is very poor and sometimes infinite. Excellent results are still obtained in the region where data are available while unstable results are obtained outside the range of extraction. The scarcity of the data is illustrated in Fig. 2.1 for two independent LTE signal in upper and lower bands.

To address the scarcity of data at high power it is possible to develop a 2D multisine with the correct CCDF which will map the complete $(|x_1|, |x_2|)$ space [1]. This is illustrated in Fig. 2.2(a), where the 2D envelopes are seen to fully map the 2D rectangular space for a more robust extraction. On the other hand, it is observed that the 2D envelopes in Fig. 2.2(b) are not fully mapping the 2D rectangular coordinate. This arises obviously from the saturation of the PA, as is also indicated by the different
Figure 2.2: Distribution of the envelopes at the PA input and output for a specially synthesized pair of multisines at LSB and USB respectively. Normalized input envelopes with an average power $|x_1|^2 + |x_2|^2$ in the same power range are plotted using the same color group on the left graph. An increment of 0.2 in normalized envelope is used. The corresponding normalized PA output envelopes are shown using the same color on (b) graph. The input and output envelopes are normalized relative to the peak envelope in each band.

color mapping for corresponding input power ranges. Note that the saturation at the output of the PA is seen to take place between the straight line $|y_1| + |y_2| = 1$ and the circle $|y_1|^2 + |y_2|^2 = 1$. Both the input and output PA data have been normalized. It results that the DPD extraction from output to input will not be fully mapped and still prone to divergence in the regions where there is no data. This problem can be resolved by the use of polar coordinates over rectangular coordinates when using splines for extracting the DPD gains.

As an alternative to the measurement of the PA in a full rectangular coordinate, the extrapolation of the gain functions can also be used. Using the saturation property $|y_1|^2 + |y_2|^2 = 1$ observed by the PA, one can deduce that the envelopes in saturation...
Figure 2.3: (a) 3D plot of envelope of $|y_1|$ and (b) $|y_2|$ at the PA output for memoryless PA model. Measured data (red dots), the extrapolated data (black circles) and the cubic-spline fit (lines) are compared

are given by:

$$|y_1| = \frac{|x_1|}{\sqrt{|x_1|^2 + |x_2|^2}} \quad \text{and} \quad |y_2| = \frac{|x_2|}{\sqrt{|x_1|^2 + |x_2|^2}}$$

(2.1)

One can easily verify that $|y_1|$ and $|y_2|$ satisfy the saturation property $|y_1|^2 + |y_2|^2 = 1$ as required. The resulting extrapolation of the gain functions is illustrated in Fig. 2.3 for the case of a memoryless PA model. The phase of $y_1$ and $y_2$ at saturation can themselves be interpolated from the data near saturation. The validity of the methodology is visually verified in Fig. 2.3 by the smooth extrapolation provided by the extrapolated data (black circles) and the extracted cubic-spline (lines) relative to the measured data (red dots). Since it is not practical to train a 2D predistorter for all possible 2D distribution of peak envelope events, ensuring for a physical extrapolation of the PA envelope response outside the range of extraction provides for a more robust 2D spline representation. This extrapolation method may help give spline
representation a competitive edge compared to the classical polynomial extraction. Splines have the ability to handle more efficiently the harder nonlinearities when the PA operates in deep compression [4]. Driving the PA in stronger compression usually yields benefit to power added efficiency (PAE).

The gain function representation and model inversion discussed in this section have provided some insights in the PA response. For example in Fig. 2.3 it is observed that the saturation of the PA took place above $|y_1| + |y_2| = 1$ and close to the circle line $|y_1|^2 + |y_2|^2 = 1$. The importance of the average envelope of $|y_1|^2 + |y_2|^2$ associated with the average power (purple line in Fig 1.8) suggests that the saturation in the PA is part of a thermal process. On the other hands, deep level traps in GaN HEMT based PA will be more affected by the peak envelope (green line in Fig 1.8) as they charge during the infrequent peaks and slowly discharge when the PA returns its operation to the average power.
CHAPTER 3

Quasi Exact Inverse

Most of the available predistorters (PDs) are based on indirect learning (IL) as shown in Fig. 3.1(a). The inverse of the PA is modeled using a postdistorter inverse model by swapping the input and output of a PA model and the coefficients are transferred to the PD. The two main drawbacks that affect the performance of this method are [26]:

1. When $y$ is noisy due to measurement setup, IL requires to find an inversion of the noisy regression matrix. Due to this the adaptive algorithm converged to biased values.

2. The identified post distorter which is copied into the PD does not guarantee a good pre-inverse filter for the nonlinear device because of using commutative property for non-linear systems.

A new scheme is developed in [27] to mitigate these issues. Initially an accurate PA model is estimated and then the DPD function is obtained by inverting the PA model as shown in Fig. 3.1(b). The DPD function is defined iteratively only for one memory delay. It takes multiple iterations for converging to the actual inverse model.
This scheme is referred as direct learning (DL) and a comparison in performance with IL is reported in [28]. It is observed that DL algorithms achieve better performance in terms of NMSE, but with a few iterations. The PD based on IL model is estimated by using a least square (LS) method and it doesn’t need any iterative process whereas the PD developed in [27] is based on iterative process. In [29], the impact of noise on the identification process of PD in indirect learning architecture is studied and verified to contribute to the degradation in NMSE value. The block diagram of the test setup for both the learning schemes is shown in Fig. 3.2. The DPD block can be residing in any DSP or FPGA device and can also include other operations like digital down converter (DDC), time alignment, and coefficient extraction.

3.1 Single Band System [2]

3.1.1 Conventional Indirect Learning

The indirect learning architecture shown in Fig. 3.1(a) is commonly used for identifying the predistorter. The PA is preceded by a pre-distorter and followed by a
post-distorter. This case uses two similar memory polynomial (MP) models for both the post-distorter and PD [30]. The DPD is implemented in two steps.

1. In the initial training, when there is no PD attached, the scaled version of output is provided as the input to the post-distorter. The coefficients are extracted using the least squares (LS) method in order to reduce the error $e$. Ideally, if we can reach $e \approx 0$ then $\tilde{z} \approx z$, creating the post-inverse for the PA.

2. Next these coefficients are copied into the DPD, which is used as pre-inverse.

The MP model used in this algorithm is defined in (3.1) where we include only the odd order terms:

$$z(n) = \sum_{k=0}^{K} \sum_{m=0}^{M} c_{km} |x(n - m)|^{2k} x(n - m)$$  \hspace{1cm} (3.1)

where $x(n - m)$ is the delayed input, $K$ is the non-linearity order, $M$ is the memory depth, $c_{km}$ are the coefficients of the model and $z(n)$ is the predistorted output from the PD.
To address the disadvantages of this indirect learning approach previously described, a new linearization algorithm is implemented wherein the PD model is extracted directly from the PA model. The performance is improved even without any iteration involved in the process owing to the use of a quasi-exact inverse for the memory polynomial/spline PA model.

3.1.2 Quasi Exact PA Inverse

An accurate modeling of the PA response is critical if one is to benefit from a QEI scheme to linearize a PA. In this work, a generalized memory PA model is used where the output is given by various nonlinear gain functions \( G_{m_a}(.) \), multiplying them with each of the delayed input excitations \( z(n - m_a) \):

\[
y(n) = \sum_{m_a=0}^{M_A} G_{m_a}(|z(n - m_a)|^2) z(n - m_a) \quad (3.2)
\]

where \( y(n) \) is the output of the PA, \( M_A \) is the PA memory depth and \( z(n - m_a) \) is the delayed input based on the index \( m_a \). \( G_{m_a} \) is the gain in each memory depth which depends on the envelop square of the input. The gains \( G_{m_a} \) can be implemented using MP, B-splines or any other function. Since the accuracy of the QEI depends on the PA model, B-splines are used here.

The DPD output can be written as given in [25]:

\[
z(n) = \sum_{m_p=0}^{M_P} \zeta_{m_p}(n) x(n - m_p) \quad (3.3)
\]

where \( z(n) \) is the output of the predistorter, \( M_P \) is the DPD memory depth and \( x(n - m_p) \) is the delayed input based on the index \( m_p \). \( \zeta_{m_p} \) is the gain of the predistorter in each memory depth.
Substituting (3.3) in (3.2) gives the overall equation for the cascaded system (PA and DPD).

\[ y(n) = \sum_{m_a=0}^{M_A} \sum_{m_p=0}^{M_P} G_{m_a}(|z(n - m_a)|^2) \zeta_{m_p}(n - m_a) x(n - m_a - m_p) \]  
(3.4)

On setting all the coefficients weighing \( x(n - m_a - m_p) \) to zero, except for the case \( m_a + m_p = m_a = m_p = 0 \), which is the linear gain of \( \alpha G_{lin}^A \) with \( \alpha \) as the targeted gain compression verifying \( 0 < \alpha < 1 \). Hence (3.12) reduces to the following system of equations:

\[
\min(k,M_A) \sum_{m_a=\max(0,k-M_P)}^{\min(k,M_A)} G_{m_a}(|z(n - m_a)|^2) \zeta_{k-m_a}(n - m_a) = \alpha G_{lin}^A \delta_k
\]
(3.5)

\[ \forall 0 \leq k \leq M_A + M_P \]

The linearization of PA model with \( M_A + 1 \) taps leads to a system of \( M_A + M_P + 1 \) equations with \( M_P + 1 \) unknown predistorter coefficients. Since this is an overdetermined system, the \( M_P + 1 \) equations are used to extract the coefficients whereas the remaining introduce the residual error. This residual error can be reduced with an increase in DPD memory depth \( M_P \). An exact solution is found when this \( M_P \) approaches infinity. However, a high convergence can be achieved with smaller number of taps. For a two tap PA model, QEI with 15 taps provides a DPD linearization of -84 dB NMSE. In practical implementation, an even smaller number of taps \( M_P \) is sufficient, given the limit in accuracy of the PA model.

The general solution with arbitrary number of taps is given as:

\[ \zeta_0(n) = \frac{\alpha G_{lin}^A}{G_0(|z(n)|^2)} \]  
(3.6)

\[ \zeta_k(n) = -\frac{1}{G_0(|z(n)|^2)} \sum_{m_a=\max(1,k-M_P)}^{\min(k,M_A)} G_{m_a}(|z(n - m_a)|^2) \zeta_{k-m_a}(n - m_a) \]  
\[ \forall 1 \leq k \leq M_P \]  
(3.7)
The solution for the memoryless case is given by (3.6), whereas the memory case is obtained from (3.7). The key feature in (3.14) and (3.15) is that the envelop $|z(n)|^2$ at the output of PD is unknown at time $n$ and must be self-consistently calculated at each new time step. From (3.3), the output of the PD is calculated using present and past values of input $x(n - m_p)$ and yet-to-be determined $\zeta_{m_p}(n)$. This $\zeta_{m_p}(n)$ depends on past values $\zeta_{m_p}(n - m_p)$ and also on the yet to be determined $|z(n)|^2$, thus creating a transcendental equation in terms of $|z(n)|^2$ that needs to be solved at each time step $n$, (3.8).

Since $S(n)$ is readily calculated using the results of (3.6) and (3.7), the unknown envelop $|z(n)|^2$ is simply obtained from the inverse of the zero delay PA model output $f^{-1}(|S(n)|^2)$. This obviously implies that the zero-delay AM-AM PA response $f(|z(n)|^2) \triangleq |G_0(|z(n)|^2)|^2 |z(n)|^2$ should be monotonous for the QEI to exist.

$$f(|z(n)|^2) \triangleq |G_0(|z(n)|^2)|^2 |z(n)|^2$$

$$= \left| \sum_{m_p=0}^{M_P} G_0(|z(n)|^2) \zeta_{m_p}(n) x(n - m_p) \right|^2 \triangleq |S(n)|^2$$

(3.8)
Figure 3.4: Measurement Results: Spectrum

### 3.1.3 Measurement Results

A new test bench close to the real base-station is developed in [4,19] for dual-band DPD. A similar setup for single band DPD as shown in Fig. 3.3 is developed here. The input data is sent from a host computer using Matlab and stored in the FPGA’s memory. The downloaded data can be played out to the ADI MSDPD board at the rate of 245.76MHz. The ADI MSDPD board integrates a complete high performance RF and a mixed-signal transmit and receive chain onto a single board. The DAC in the transmit path is programmed with a sampling frequency of 983.04MHz i.e. with an interpolation of 4. The signal is upconverted to 2200MHz using an I/Q modulator and sent to a 10W NXP PA. The amplification stage is composed of a cascade of 1W Prewell linear driver followed by a broadband (500-2500MHz) 10W
Figure 3.5: Measurement Results: In/Out, AM/AM and AM/PM curves of the PA
peak output power PA, based on the NXP Semiconductor GaN HEMT CLF1G0060-10 transistor [31] biased in Class-AB (Vds = 50V and Ids = 40mA). At 2 GHz, the output power for 1dB gain compression is 36 dBm and the drain-efficiency is \( \eta_D = 21\% \). The output from the PA is fed back to the observation path on the MSDPD board. The signal is downconverted to an 184.32 MHz IF signal, digitized and stored in the FPGA’s memory. The stored data can be used for further processing like digital down conversion (DDC), time alignment and extracting the coefficients for the DPD system. An approximation of the inverse model is determined using the PA model as described in Section 3.1.2.

The PA in the setup experiences a compression of 1.6 dB as shown in Fig. 3.5(a). In MP-Indirect learning architecture, \( M = 2 \) and \( N = 7 \) are used for PD model, whereas \( M_A = 1 \) and \( M_P = 11 \) are used in the QEI model. In both cases, 24 coefficients are used for the PD model. The B-spline model used for the PA relies on 10 spline intervals over the entire envelop range. The PA models are extracted in both case using 8192 samples and verified for 65536 samples. No iteration is used, since in real communication systems, the same waveform broadcast is never played twice while the PD is continuously trained. It can be observed from Fig. 3.5(b) that there is very little memory effect in the PA, but the MP-PD has to use higher memory depth for better performance when compared to QEI PA model. The performance of QEI exhibits a noticeable improvement over the MP-PD as shown in Table. 3.1 in terms ACPR and NMSE. The comparison of spectrum at the PA output for both the models are shown in Fig. 3.4. An improved performance is expected for waveforms driving the PA further in compression. Further improvements could also be achieved using
an adaptive scheme to more frequently update the PA model as the waveform and PA evolve in time.

### Table 3.1: Comparison of NMSE and ACPR for Single Band System

<table>
<thead>
<tr>
<th>Configuration</th>
<th>NMSE (dB)</th>
<th>ACPR (dBc) @±5 MHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Without DPD</td>
<td>-13.6</td>
<td>-38.9 / -38.4</td>
</tr>
<tr>
<td>MP: Indirect Learning</td>
<td>-36.5</td>
<td>-54.6 / -53.8</td>
</tr>
<tr>
<td>Quasi Exact Inverse</td>
<td>-38.8</td>
<td>-56.2 / -56.9</td>
</tr>
</tbody>
</table>

### 3.2 Dual band System [3]

The generalized architecture of a dual band transmitter using Frequency selective technique is shown in Fig. 3.6. Most of the available dual-band DPD’s are based on IL as shown in Fig. 3.7, wherein the inverse of the PA is modeled using a postdistorter inverse model and the coefficients are transferred to the PD.
The architecture used for DL is shown in Fig. 3.8, which is extended version of single band DL architecture [2] to dual-band DL system. A new DL technique using the quasi-exact inverse [2,25] of the PA model is developed for single band case in Section 3.1, showing performance improvement in terms of ACPR and NMSE when compared to IL. In this section, the same principle is extended to dual-band case to incorporate these benefits into the dual-band DPD system.

### 3.2.1 Indirect Learning

This architecture uses two similar MP models for both the post-distorter and PD. The MP model used in this algorithm is defined in (3.9) where we include only the odd order terms [17]:

\[
z_i(n) = \sum_{m=0}^{M-1} \sum_{k=0}^{N-1} \sum_{j=0}^{k} c_{jkm}^{(i)} |x_i(n-m)|^{2k} |x_i(n-m)|^{2(k-j)} x_i(n-m)
\] (3.9)
where ∀ i, l ∈ [1, 2] and i ≠ l, x_i(n - m) is the delayed input, K is the non-linearity order, M is the memory depth, c_{jm}^{(i)} are the coefficients of the model and z_i(n) is the predistorted output from the PD.

### 3.2.2 2D Quasi Exact Inverse

A generalized memory PA model can be given as:

\[
y_i(n) = \sum_{m_a=0}^{M_A-1} G_{i,m_a} (|z_i(n - m_a)|^2, |z_l(n - m_a)|^2) z_i(n - m_a)
\]  

(3.10)

where |z_i(n - m_a)|^2 is envelop squares of each band with memory, G_{i,m_a}(.) and G_{2,m_a}(.) are the gain functions for each band. The gain functions with memory are functions of the envelop squares of each band. The gains G_{i,m_a}, can be implemented using memory

Figure 3.8: Quasi Exact Inverse
polynomial, splines or any other functions. In order to compare the performance
between the DL and IL in the presence of noise, 2D memory polynomials are used
here.

The DPD output can be written as given in (3.11):

\[ z_i(n) = \sum_{m_p=0}^{M_P-1} \zeta_{i,m_p}(n)x_i(n-m_p) \tag{3.11} \]

where \( M_P \) is the DPD memory depth and \( \zeta_{i,m_p} \) are the gain of the predistorters for
each memory depth.

Substituting (3.11) in (3.10) gives the overall equation for the cascaded system
(PA and DPD):

\[ y_i(n) = \sum_{m_a=0}^{M_A-1} \sum_{m_p=0}^{M_P-1} G_{i,m_a} (|z_i(n-m_a)|^2, |z_i(n-m_a)|^2) \zeta_{i,m_p}(n-m_a) \cdot x_i(n-m_a-m_p) \tag{3.12} \]

On setting all the coefficients weighing \( x_i(n-m_a-m_p) \) to zero, except for the case
\( m_a + m_p = m_a = m_p = 0 \), which is the linear gain of \( \alpha_i G_{i,A}^{\text{lin}} \) with \( \alpha_i \) as the targeted
gain compression verifying \( 0 < \alpha_i < 1 \). Hence (3.12) reduces to the following system
of equations:

\[ \min_{k,M_A} \sum_{m_a=\max(0,k-M_P)} G_{i,m_a} (|z_1(n-m_a)|^2, |z_2(n-m_a)|^2) \zeta_{i,k-m_a}(n-m_a) = \alpha_i G_{i,A}^{\text{lin}} \delta_k \quad \forall 0 \leq k \leq M_A + M_P \tag{3.13} \]

The linearization of PA model with \( M_A + 1 \) taps leads to a system of \( M_A + M_P + 1 \)
equations with \( M_P + 1 \) unknown predistorter coefficients. Since this is an overde-
termined system, the \( M_P + 1 \) equations are used to extract the coefficients whereas the
remaining introduce the residual error. This residual error can be reduced with an
increase in DPD memory depth $M_P$. Ideally an exact solution is found when this $M_P$ approaches infinity. The general solution with arbitrary number of taps is:

$$\zeta_{i,0}(n) = \frac{\alpha_i G_{i,\lambda}^{lin}}{G_{i,0}(|z_1(n)|^2, |z_2(n)|^2)}$$

$$\zeta_{i,k}(n) = -\frac{1}{G_{i,0}(|z_1(n)|^2, |z_2(n)|^2)} \sum_{m_a = \max(1,k-M_P)}^{\min(k,M_A)} G_{i,m_a}(|z_1(n-m_a)|^2, |z_1(n-m_a)|^2) \cdot \zeta_{i,k-m_a(n-m_a)}$$

$$\forall 1 \leq k \leq M_P$$

The solution for the memoryless case is given by (3.14), whereas the memory case is obtained from (3.15). The key feature in (3.14) and (3.15) is that the envelop $|z_1(n)|^2$ and $|z_2(n)|^2$ at the output of both the PD’s are unknown at time $n$ and must be self-consistently calculated at each new time step using (3.16).

$$f_i(|z_1(n)|^2, |z_2(n)|^2) \triangleq |G_{i,0}(|z_1(n)|^2, |z_2(n)|^2)|^2 |z_i(n)|^2$$

$$= \left| \sum_{m_p=0}^{M_P} G_{i,0}(|z_1(n)|^2, |z_2(n)|^2) \zeta_{i,m_p}(n) x_i(n-m_p) \right|^2$$

Since $S_i(n)$ is readily calculated from (3.14) and (3.15), the unknown envelopes $\langle |z_1(n)|^2, |z_2(n)|^2 \rangle$ are obtained from the inverse of the zero delay PA model output $f_i^{-1}(|S(n)|^2)$. This implies that for the 2D-QEI to exist, each zero-delay AM-AM PA outputs $[f_1(|z_1(n)|^2, |z_2(n)|^2), f_2(|z_1(n)|^2, |z_2(n)|^2]$ must originate from single $[|z_1(n)|^2, |z_2(n)|^2]$ input pair. This requirement is usually satisfied as long as the PA is not over-saturated.
3.2.3 Measurement Results

The test bench setup is shown in the Fig. 3.9. The test bench is similar to the setup that is used in Section 3.1, except a second MSDPD is used to transmit and receive the concurrent band. The frequency spacing between the bands is around 310 MHz, where the lower band is transmitted at 1890 MHz and the upper band is at 2200 MHz. A reference clock (refclk) is used for synchronization between the boards as shown in Fig 3.9. The output from the PA is fed back to the observation path on both the boards. The signal is downconverted to an 184.32 MHz IF signal and stored in the FPGA’s memory.

Table 3.2: Comparison of NMSE and ACPR for -50 dBc Noise Floor for Dual-Band system

<table>
<thead>
<tr>
<th>Configuration</th>
<th>NMSE(dB)</th>
<th>ACPR (dBc) @±5 MHz</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LSB</td>
<td>USB</td>
</tr>
<tr>
<td>Without DPD</td>
<td>-20.99</td>
<td>-21.94</td>
</tr>
<tr>
<td>Indirect Learning</td>
<td>-34.43</td>
<td>-33.69</td>
</tr>
<tr>
<td>QEI</td>
<td>-38.43</td>
<td>-37.42</td>
</tr>
</tbody>
</table>
A 2D MP with non-linearity order (N=7) and memory length (M=2) are used in both the DL and IL model extractions. $M_P = 10$ is used for calculating the QEI. No iteration is used for both the DL and IL DPD extractions, which are performed in a single step like in practical systems. The spectra without and with DPD for additive white noise in the receiver yielding a -50 dBc noise floor, are shown in Fig. 3.10. It can be observed from Fig. 3.10 and Table 3.2 that in the presence of additive white noise, the direct learning architecture using QEI performs better than the indirect learning architecture in terms of NMSE and ACPR by up to 4 dB.
A polynomial model of limited order is commonly used to represent the gains of a PA for the entire input envelope range [17, 32, 33]. A polynomial expansion with higher order is required to represent PAs which are highly non-linear and saturated. However, extracting such a polynomial based model can lead to a numerically ill-conditioned regression matrix and yields to a highly oscillatory solution when the data is sparse [34]. Due to this ill-conditioning, the coefficients are extremely sensitive to the data [19] thereby forcing to update the system even if there is small error in reading at the input data. The coefficients also have a global scope, which tend to effect the entire curve and thus increase the error. A piece-wise polynomial representation like C-spline may be considered instead of regular polynomials to remediate these issues as is discussed next.

It may become profitable to rely on piece-wise polynomials of lower order rather than high-order polynomials for the entire envelope range to provide for a more robust representation. Since lower order polynomials are used, they do not have oscillations like the Runge phenomenon [35]. The condition number of the matrix created using piece-wise polynomials will also be smaller than regular polynomials, thereby making the system more stable to errors at the input. The C-spline coefficients have a
local scope, i.e. when an approximate value is used for a particular knot, the error introduced will only create a local distortion in the shape of the spline which does not propagate to the entire input range [36]. As a result, a more robust extraction is expected for linearizing subsequent data after the training is performed.

4.1 Single-Band System

When amplifying narrow band signals, the PAs can be assumed to be memoryless and a memoryless DPD can be used to linearize them. However, this assumption is no longer valid for the wide-band and high PAPR communication signals which are currently in use. Implementing DPD which takes into account time-selective memory effects [8] gives a considerable improvement in the ACPR and NMSE [34].
4.1.1 Least Square Cubic Spline

The general frequency and time-selective memory model used to represent the output of the PA is as follows:

\[ y(n) = \sum_{m=0}^{ML-1} G(m)(|x(n-m)|^2)x(n-m) \]  \hspace{1cm} (4.1)

where \( |x(n-m)|^2 \) is the envelope squares of input with memory and \( ML \) the memory depth. The gain function \( G(m)(|x|^2) \) for the memory delay index \( m \) is function of the envelop squares of the input. These gain function can be represented using different basis functions such as conventional polynomials [10], orthogonal polynomials [18] or splines [34]. Here new 1D C-splines are used to represent these gains.

The gain function for each memory delay index \( m \) in (4.1) can be represented using the following new 2D basis functions \( \phi_i(|x|^2) \) as:

\[ G(m)(|x|^2) = \sum_{i=0}^{N_s1} c_i^{(m)} \phi_i(|x|^2) \]  \hspace{1cm} (4.2)

where \( c_i^{(m)} \) is the weight coefficients of the 1D basis of the single band.

The \( \phi_i \) functions for the case of a 4 equally spaced knot grid, are plotted in Fig. 4.2. It can be observed that the \( \phi_i \) functions decay rapidly from 1 at \((u)\) to 0 at all the other knots over the entire amplitude range. The cubic splines provide continuity up to second order derivatives between the knots.

Similar to a Lagrange interpolation polynomial, the 1D \( \phi_i(|x|^2) \) along the \(|x|^2\) is represented using 1D C-spline with \((N_s)\) number of spline as:

\[ \phi_i\left(|x_u|^2\right) \triangleq \delta_{i,u} \]

\[ = \begin{cases} 
1 & \forall \ i = u \\
0 & \forall \text{ other knots} 
\end{cases} \]  \hspace{1cm} (4.3)

\[ \forall \ i \in [0, N_s], \ \forall \ u \in [0, N_s - 1] \]
In the knot interval $[|x_u|^2, |x_{u+1}|^2]$ the 1D $\phi_i^{(v)}(|x|^2)$ using 1D C-spline is then:

$$\phi_i^{(u)}(|x|^2) = \sum_{q=0}^{3} c_i^{(iu)} (|x|^2 - |x_u|^2)^q$$

\(\forall \ i \in [0, N_s], \ \forall \ u \in [0, N_s - 1]\)  

(4.4)

The $\phi_i$ functions can be represented in a vector form (4.5) where each element is computed from a set of splines defined along the entire envelop region:

$$\Phi_i(|x|^2) = [\phi_0(|x|^2), \phi_1(|x|^2), \cdots, \phi_{N_s}(|x|^2)]$$

(4.5)

Substituting this basis function for the gain in (4.1) will give the outputs as (4.6):

$$y(n) = \sum_{m=0}^{ML-1} x(n - m) \sum_{i=0}^{N_s} c_i^{(m)} \phi_i(|x(n - m)|^2)$$

(4.6)

From (4.6) the coefficients can be extracted using least square method [10]. The advantage of the proposed C-spline basis is that the data used for the extraction of the $c_i^{(m)}$ coefficients can randomly span the complete region of $|x|^2$.  

49
4.1.2 Measurement Results

A similar setup which is used in Section 3.1.3 is also used here to verify the new LSCS for single band system. For comparison, a memory polynomial is used in indirect learning architecture to extract the coefficients. A 3c-WCDMA signal is used for performance comparison. The basis functions are created using cubic spline with 5 knots. There are 5 basis functions considered in this measurement setup and a memory depth of 3 is needed to include the memory effect of the power amplifier. Thereby there are 15 coefficients computed for entire model. A non-linearity order (NL) of 5 and memory length of 3 are used for polynomial model so that they use the same number of coefficients as LSCS. In can be observed from Fig. 4.3 that, for the same number of coefficients, LSCS achieves better performance in terms of NMSE and ACPR as tabulated in Table. 4.1

<table>
<thead>
<tr>
<th>Configuration</th>
<th>NMSE (dB)</th>
<th>ACPR (dBc) @±5 MHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Without DPD</td>
<td>-14.8</td>
<td>-38.9 / -38.4</td>
</tr>
<tr>
<td>Polynomial</td>
<td>-39.4</td>
<td>-49.8 / -50.7</td>
</tr>
<tr>
<td>LSCS</td>
<td>-41.3</td>
<td>-51.0 / -53.2</td>
</tr>
</tbody>
</table>

4.2 Dual-Band System

The general frequency and time-selective memory model used to represent the output of the PA in each band after the out-of-band IMD’s are filtered is as follows:
\[
y_1(n) = \sum_{m=0}^{ML-1} G_1^{(m)}(|x_1(n - m)|^2, |x_2(n - m)|^2)x_1(n - m)
\]
\[
y_2(n) = \sum_{m=0}^{ML-1} G_2^{(m)}(|x_1(n - m)|^2, |x_2(n - m)|^2)x_2(n - m)
\] (4.7)

where \(|x_1(n - m)|^2\) and \(|x_2(n - m)|^2\) are the envelope squares of each frequency band with memory and \(ML\) the memory depth. The gain functions \(G_1^{(m)}(|x_1|^2, |x_2|^2)\) and \(G_2^{(m)}(|x_1|^2, |x_2|^2)\) for the memory delay index \(m\) are functions of the envelope squares of both the bands. These gain functions can be represented using different basis functions such as conventional polynomials [17], orthogonal polynomials [19] or splines [4]. In this chapter 2D C-splines are used to represent these gains. Two methods are developed to design a 2D C-spline DPD, and their performance and limitations are discussed.
4.2.1 2D Conventional Cubic Spline (2D-CS) [4]

Extraction

The generation of a 2D-CS starts with the synthesis of multiple 1D C-splines which are then extended to 2D. A 1D C-spline consists of piece-wise polynomials of order 3 which are continuous at the knots where the splines connect, not only for the polynomials themselves, but also for their first and second derivatives [4,35]. The 2D gain function for different memory delay indices $m$ is shown in Fig.4.1 with $G_p^{(m,u,v)}(|x_1|^2,|x_2|^2)$ representing the 2D C-spline for the particular region indices $(m,u,v)$. The entire 2D space is divided into a grid structure with the grid intersections representing the knots $(u,v)$. The number of splines along each axis $|x_1|^2$ and $|x_2|^2$ is defined as $N_{s_1}$ and $N_{s_2}$.

The construction of these 2D C-splines(2D-CS) is performed as follows:

(a) Firstly, start with the extraction of 1D C-splines along $|x_1|^2$ axis as shown in Fig.4.1, so that the gain function in (4.7) can be written for a given memory delay index $m$ and frequency band $p=[1,2]$ as:

$$G_p^{(m,u,v)}(|x_1|^2,|x_2|^2) = \sum_{i=0}^{3} b_{pi}^{(muv)}(|x_2|^2)(|x_1|^2-|x_{1,u}|^2)^i$$

$$\forall \ u \in [0, N_{s_1} - 1], \forall \ v \in [0, N_{s_2} - 1]$$

and $\forall \ m \in [0, ML - 1]$ (4.8)

where $G_p^{(m,u,v)}(|x_1|^2,|x_2|^2)$ is the gain function represented by a 2D C-spline for the intersecting area spanned by $[u,u+1]$ along $|x_1|^2$ and $[v,v+1]$ along $|x_2|^2$ as shown in Fig.4.1. $|x_{1,u}|^2$ are the knot values along $|x_1|^2$ axis given by $[|x_{1,0}|^2 \ |x_{1,1}|^2 \ldots |x_{1,N_{s_1}}|^2]$. The coefficients $b_{pi}^{(muv)}(|x_2|^2)$ which are extracted in
this step for all the splines $i$ along $|x_1|^2$ are themselves a function of the $|x_2|^2$ parameter.

(b) Next, 1D C-splines are extracted for the extracted coefficients in (4.8) along $|x_2|^2$. It can be observed that the coefficients $b^{(muv)}_{pi}(|x_2|^2)$ from (4.8) are used to construct the 1D C-splines in (4.9):

$$b^{(muv)}_{pi}(|x_2|^2) = \sum_{j=0}^{3} a^{(muv)}_{pij} (|x_2|^2 - |x_{2,v}|^2)^j \quad (4.9)$$

where $|x_{2,v}|^2$ are the knot values along the $|x_2|^2$ axis, given by $[|x_{2,0}|^2, |x_{2,1}|^2, \ldots, |x_{2,N_{x2}}|^2]$. $a_{pij}^{(muv)}$ are the C-spline coefficients for the intersecting knot area spanned by $[u, u+1]$ along $|x_1|^2$ and $[v, v+1]$ along $|x_2|^2$.

(c) Finally the 2D-CS can be constructed by substituting (4.9) in (4.8):

$$G^{(m,u,v)}_{p}(|x_1|^2, |x_2|^2) = \sum_{i=0}^{3} \sum_{j=0}^{3} a^{(muv)}_{pij} (|x_1|^2 - |x_{1,u}|^2)^i \cdot (|x_2|^2 - |x_{2,v}|^2)^j \quad (4.10)$$

The result is equivalent to a tensor product of univariate C-spline interpolation as shown in (4.10).

**Implementation**

The architecture for the 2D-CS is shown in Fig.4.4(a) with the implementation of C-spline as shown in Fig.4.4(b). Each branch corresponds to a memory delay index $m$ to take into account the time-selective memory effect of the PA. The extracted coefficients can be stored in a 2D-LUT structure as shown in Fig.4.4(b). The coefficients of the 2D-CS which are given by $a^{(muv)}_{pij}$ in (4.10) are used to interpolate the gain values at the value of $|x_1(n-m)|^2$ and $|x_2(n-m)|^2 \forall m$. 

53
To implement the 2D-LUT, the entire LUT structure can be divided into a sequence of multiple 1D-LUTs as shown in the Fig.4.4(b). Each coefficient of the cubic polynomial is stored in separate 1D-LUTs and the address for each coefficient is calculated based on the value of $|x_1(n-m)|^2$ and $|x_2(n-m)|^2$. The normalized envelope square of each band is multiplied with the number of splines to get the address along each axis. Using these values one can calculate the effective 2D address as (4.11) as follows:

$$addr_1 = \text{floor} \left\{ |x_1(n-m)|^2 \cdot N_{s1} \right\}$$

$$addr_2 = \text{floor} \left\{ |x_2(n-m)|^2 \cdot N_{s2} \right\}$$

$$addr_{a_{pij}} = addr_2 + addr_1 \cdot N_{s2}$$

(4.11)

where $addr_1$ and $addr_2$ defines the address in each direction respectively, and $addr_{a_{pij}}$ is the compounded 2D address for the 2D C-spline coefficients $a_{pij}^{(muv)}$ for any arbitrary $|x_1(n-m)|^2$ and $|x_2(n-m)|^2$ values. Using this address the coefficients can be selected from the LUT and can be used to calculate the respective gains $G_p^{(m,u,v)}(|x_1(n-m)|^2, |x_2(n-m)|^2)$ using (4.10). In terms of computation when using a fixed point arithmetic, the floor function is effectively implemented by using the correct number of most significant bits and the address mapper just requires one multiplication and one addition.

With this conventional 2D C-spline approach, a major limitation of using 2D-CS is the necessity to first evaluate the gains $G_p^{(m,u,v)}(|x_1|^2, |x_2|^2)$ at the knots $(|x_{1,u}|^2, |x_{2,v}|^2)$ using a separate modeling method before proceeding to the extraction of the C-spline coefficients $a_{pij}^{(muv)}$. The C-spline coefficients must indeed be extracted at the knots to enforce exactly the continuity of the second order derivatives. Note also that this
model requires \(2ML \cdot N_{s1} \cdot N_{s2} \cdot 16\) coefficients, which grows rapidly with an increase in the number of splines along each axis.

### 4.2.2 2D Least Square Cubic Spline (2D-LSCS)

The performance of the conventional 2D-CS reviewed in section 4.2.1 is limited by the preliminary gain extraction, typically using 2D polynomials [4], which is required to first determine the gains at the knots. Furthermore, this method also requires a large number of coefficients. To overcome these deficiencies a new C-spline basis representation is developed in this section for which the coefficients can be extracted directly from the measured data using the least square (LS) method. This C-spline basis is designed to exhibit the desired local focus discussed in section 4.2.1 while also being sensitive to the data distribution via the selection of a non-uniform knot distribution.

**Extraction**

The gain function for each memory delay index \(m\) in (4.7) can be represented using the following new 2D basis functions \(\phi_{ij}(|x_1|^2, |x_2|^2)\) as:

\[
G_1^{(m)}(|x_1|^2, |x_2|^2) = \sum_{i=0}^{N_{s1}} \sum_{j=0}^{N_{s2}} c_{1ij}^{(m)} \phi_{ij}(|x_1|^2, |x_2|^2)
\]

\[
G_2^{(m)}(|x_1|^2, |x_2|^2) = \sum_{i=0}^{N_{s1}} \sum_{j=0}^{N_{s2}} c_{2ij}^{(m)} \phi_{ij}(|x_1|^2, |x_2|^2)
\]  

(4.12)

where \(c_{1ij}^{(m)}\) and \(c_{2ij}^{(m)}\) are the weight coefficients of the 2D basis for band 1 and 2 respectively. The basis functions \(\phi_{ij}(|x_1|^2, |x_2|^2)\) are created using 2D C-splines so that they verify a 2D Kronecker delta property (4.13), in which the value of \(\phi_{ij}\) at the
Figure 4.4: 2D-CS implementation with memory for each band $i$. 
2D knots values \((|x_{1,u}|^2, |x_{2,v}|^2)\) is equal to one when \((i, j) = (u, v)\) and zero elsewhere:

\[
\phi_{ij}(|x_{1,u}|^2, |x_{2,v}|^2) \triangleq \delta_{i,u} \cdot \delta_{j,v} = \begin{cases} 
1 & \forall (i, j) = (u, v) \\
0 & \forall \text{other knots}
\end{cases}
\]

\(\forall i \in [0, N_{s1}], \forall j \in [0, N_{s2}]\)

\(\forall u \in [0, N_{s1} - 1], \forall v \in [0, N_{s2} - 1]\)

(4.13)

The \(\phi_{ij}\) functions for the case of a 4 by 4 equally spaced knot grid, are plotted in Fig. 4.5. It can be observed that the \(\phi_{ij}\) functions decay rapidly from 1 at \((u, v)\) to 0 at all the other knots over the entire amplitude range. The cubic splines provide continuity up to second order derivatives between the knots. In this work the 5 by 5 knot distribution for both \((|x_{1,u}|^2\) and \(|x_{2,v}|^2)\) is equal to \([0, 1/4, 1/2, 3/4, 1]^2 = [0, 1/16, 1/4, 9/16, 1]\) so that the allocation of the C-spline basis degrees of freedom better reflects the data distribution.

The 2D \(\phi_{ij}\) functions are synthesized from the 1D \(\phi_i\) and \(\phi_j\) functions along \(|x_1|^2\) and \(|x_2|^2\) directions respectively as described below.

(i) Similar to a Lagrange interpolation polynomial, the 1D \(\phi_i(|x_1|^2)\) along the \(|x_1|^2\) is represented using 1D C-spline with:

\[
\phi_i (|x_{1,u}|^2) \triangleq \delta_{i,u} = \begin{cases} 
1 & \forall i = u \\
0 & \forall \text{other knots}
\end{cases}
\]

\(\forall i \in [0, N_{s1}], \forall u \in [0, N_{s1} - 1]\)

(4.14)

In the knot interval \([|x_{1,u}|^2, |x_{1,u+1}|^2]\) the 1D \(\phi_i^{(u)}(|x_1|^2)\) using 1D C-spline is then:

\[
\phi_i^{(u)} (|x_1|^2) = \sum_{q_1=0}^{3} c_{q_1}^{(iu)} (|x_1|^2 - |x_{1,u}|^2)^{q_1}
\]

\(\forall i \in [0, N_{s1}], \forall u \in [0, N_{s1} - 1]\)

(4.15)
The $\phi_i$ functions can be represented in a vector form (4.16) where each element is computed from a set of splines defined along the entire envelop region:

$$\Phi_i (|x_1|^2) = [\phi_0 (|x_1|^2), \phi_1 (|x_1|^2), \cdots, \phi_{N_{s1}} (|x_1|^2)]$$  \hspace{1cm} (4.16)

(ii) Similarly, for the 1D $\phi_j(|x_2|^2)$ along the $|x_2|^2$ is represented using 1D C-spline with:

$$\phi_j (|x_2, v|^2) \triangleq \delta_{j,v} = \begin{cases} 1 & \forall \ j = v \\ 0 & \forall \ other \ knots \end{cases}$$  \hspace{1cm} (4.17)

In the knot interval $[|x_2, v|^2, |x_2, v+1|^2]$ the 1D $\phi_j^{(v)}(|x_2|^2)$ using 1D C-spline is then:

$$\phi_j^{(v)} (|x_2|^2) = 3 \sum_{q_2=0}^{3} e_{q_2}^{(jv)} (|x_2|^2 - |x_2, v|^2)^{q_2}$$  \hspace{1cm} (4.18)

$$\forall \ j \in [0, N_{s2}], \forall \ v \in [0, N_{s2} - 1]$$

The $\phi_j$ functions can be represented in the vector form (4.19) with each column corresponding to a spline along the entire envelop region:

$$\Phi_j (|x_2|^2) = [\phi_0 (|x_2|^2), \phi_1 (|x_2|^2), \cdots, \phi_{N_{s2}} (|x_2|^2)]$$  \hspace{1cm} (4.19)

(iii) The tensor product of (4.15) and (4.18) generates the 2D $\phi_{ij}$ functions such that in each of the 2D knot regions defined by the corners $[|x_{1,u}|^2, |x_{2,v}|^2]$ and $[|x_{1,u+1}|^2, |x_{2,v+1}|^2]$, the 2D $\phi_{ij}^{(uv)}(|x_1|^2, |x_2|^2)$ is then:

$$\phi_{ij}^{(uv)} (|x_1|^2, |x_2|^2) = 3 \sum_{q_1=0}^{3} \sum_{q_2=0}^{3} e_{q_1}^{(iu)} \cdot e_{q_2}^{(jv)} \cdot (|x_1|^2 - |x_{1,u}|^2)^{q_1} \cdot (|x_2|^2 - |x_{2,v}|^2)^{q_2}$$  \hspace{1cm} (4.20)

$$\forall \ i \in [0, N_{s1}], \forall \ j \in [0, N_{s2}]$$

$$\forall \ u \in [0, N_{s1} - 1], \forall \ v \in [0, N_{s2} - 1]$$  \hspace{1cm} (4.21)
It can be verified from (4.23) that this new C-spline basis can be used to model a flat surface without any ripple:

\[ \sum_{i=0}^{N_{s1}} \sum_{j=0}^{N_{s2}} \phi_{ij}(|x_1|^2, |x_2|^2) = 1 \quad (4.23) \]

\[ \forall |x_1|^2, |x_2|^2 \]

The memoryless 2D function \( \phi_{ij} \) can be represented in a vector form similar to a Kronecker product as (4.24). It can be observed that the structure is similar to the polynomial model, but it is created using the basis functions generated using C-splines:

\[ \Phi_{ij}(|x_1|^2, |x_2|^2) = \{ \]

\[ \left[ \phi_0(|x_1|^2) \cdot \phi_0(|x_2|^2), \phi_0(|x_1|^2) \cdot \phi_1(|x_2|^2), \ldots, \phi_0(|x_1|^2) \cdot \phi_{N_{s2}}(|x_2|^2) \right] \]

\[ \left[ \phi_1(|x_1|^2) \cdot \phi_0(|x_2|^2), \phi_1(|x_1|^2) \cdot \phi_1(|x_2|^2), \ldots, \phi_1(|x_1|^2) \cdot \phi_{N_{s2}}(|x_2|^2) \right] \]

\[ \ldots \quad \ldots \quad \ldots \]

\[ \left[ \phi_{N_{s1}}(|x_1|^2) \cdot \phi_0(|x_2|^2), \phi_{N_{s1}}(|x_1|^2) \cdot \phi_1(|x_2|^2), \ldots, \phi_{N_{s1}}(|x_1|^2) \cdot \phi_{N_{s2}}(|x_2|^2) \right] \]

\[ = \Phi_i(|x_1|^2) \otimes \Phi_j(|x_2|^2) \]

The total number of complex coefficients used in this model are \( ML \cdot (N_{s1} + 1) \cdot (N_{s2} + 1) \) which is reduced by a factor 16 compared to 2D-CS.

Substituting this basis function for the gain in (4.7) will give the outputs as (4.24):

\[ y_1(n) = \sum_{m=0}^{ML-1} x_1(n-m) \sum_{i=0}^{N_{s1}} \sum_{j=0}^{N_{s2}} c_{1ij}^{(m)} \phi_{ij}(|x_1(n-m)|^2, |x_2(n-m)|^2) \]

\[ y_2(n) = \sum_{m=0}^{ML-1} x_2(n-m) \sum_{i=0}^{N_{s1}} \sum_{j=0}^{N_{s2}} c_{2ij}^{(m)} \phi_{ij}(|x_1(n-m)|^2, |x_2(n-m)|^2) \quad (4.24) \]
Figure 4.5: 2D basis functions $\phi_{ij}$ of LSCS for each knot along $|x_1|^2$ and $|x_2|^2$.

The advantage of the proposed C-spline basis is that the data used for the extraction of the $c^{(m)}_{1ij}$ and $c^{(m)}_{2ij}$ coefficients can now randomly span the complete continuous ($|x_1|^2, |x_2|^2$) 2D space for dual bande PA modeling or ($|y_1|^2, |y_2|^2$) 2D space for dual band PA linearization. It results that the 2D-LSCS coefficients can now be directly extracted from the measured PA data using the LS method. In the conventional 2D-CS approach the data required for the extraction were the discrete C-spline knots and a direct extraction from the measured PA data was not possible.
FPGA Implementation

There can be multiple approaches for the implementation of 2D-LSCS in an FPGA. The merits and limitations of the resulting FPGA architectures are analyzed below:

(a) The spline coefficients which are used to construct the $\phi_{ij}$ in (4.24) and weighted coefficients can be stored in memory and (4.24) is constructed using adders, multipliers and delay blocks for values of $|x_1(n-m)|^2$ and $|x_2(n-m)|^2$. This kind of implementation requires a lot of resources thereby limiting the number of splines and also the memory depth of the model that can be implemented for a targeted FPGA. Due to the large data processing required, the latency of the model will also increase which will in turn increase the adaptation time for the model for a particular signal.

(b) A 2D LUT implementation can be used to store the cubic spline basis functions $\phi_{ij}^{(w)}(|x_1|^2, |x_2|^2)$ versus discrete values of $|x_1|^2$ and $|x_2|^2$ [37]. Given the values of $|x_1(n-m)|^2$ and $|x_2(n-m)|^2$ the gain value can then be extrapolated and used in (4.7). This implementation is quite general and can be used with other representation techniques extracted using different basis functions. Although this architecture benefits from a reduced latency, it requires a large amount of memory beyond the reach of conventional FPGA.

(c) A hybrid architecture can be used to compromise between the FPGA’s resource and memory utilization [38]. The 1D basis functions are precomputed and stored in memory in 1D LUTs. The 2D basis function is computed from these 1D basis functions inside the FPGA using a reduced number of multipliers. The latency is
reduced compared to (a) due to the precomputed LUTs, and since 1D functions are used instead of 2D functions, it reduces the memory storage compared to (b). The weighted coefficients are stored in registers for multiplying with the basis functions to compute the final gain values. Since the LUT can be populated with any desired 1D basis functions, this architecture can be used by other types of algorithms such as 2D memory polynomials.

A memoryless gain function is implemented using the hybrid architecture as shown in Fig. 4.6. The LUTs shown in Fig. 4.6 are used to store the 1D basis functions at each knot over the entire amplitude range. The same LUT memory can be used for both of the bands since the same basis functions are used, hence all the LUTs have two input and two outputs. A 2D basis function is calculated using the tensor

Figure 4.6: 2D-LSCS basis computation
product of the 1D basis as shown in Fig. 4.7. The weighting coefficients are then multiplied with the LUT basis values to obtain the gain value. Once the gain value is computed, the top level architecture is similar to the one shown in Fig.4.4(a). Note that the same latency is obtained for all basis functions.

The clock architecture of the entire setup is shown in Fig. 4.8. A system clock of 61.44 MHz is used to synchronize between the two MSDPD boards. A network clock from SMA port of one of the MSDPD board is used as the reference clock for the DDR3 memory inside the FPGA. An internal clock X6 can also be used as the reference clock, but it is not used here to simplify the design. The phase locked loop (PLL) inside the DDR3 generates the reference clock for the PLL inside the DAC interface (dac_ifc). The ADC interface module creates the data format for ADC using
the DDR IO block which operates on both the raising and falling edge. The dac_pll also generated the dac_clk for aligning the clock with the data. The PLL inside the MSDPD generate necessary clock frequencies like ADC clock, DAC clock, network clock. The ADC clock is also sent to the ADC interface (adc_ifc) through the HSMC pin as a reference to the PLL inside the adc_ifc. The data is packed and stored inside the DDR using the write DMA.

The transmit local oscillator (LO) and receive LO both share the same LO in order to reduce phase noise of the received signal. The DAC interpolates to 983.04 MSPS and applies a frequency translation of IF, 184.32 Mhz to the data stream. Even though zero-IF can be used, using complex IF shifts the main signal away from dc where LO feedthrough and images can be easily filtered. The observation path consists of mixer which is responsible for directly mixing the observed RF signal to a suitable IF. The typical IF frequency used in the application is 184.32 MHz. The received signal is digitized using a 245.76 MSPS ADC.

4.2.3 Measurement Results

The input data is sent from a host computer and stored in the FPGA’s double data rate (DDR) memory using Matlab and a universal serial bus (USB) connection. The downloaded data can be played out to a analog device’s mixed signal DPD (ADI MSDPD) board at the rate of 245.76 MHz. The ADI’s MSDPD board integrates a complete high performance RF and mixed-signal transmit and receive chain onto a single board. A full observation path that accepts the sampled RF output of upto 16dbm and mixes it down to suitable IF frequency that is digitized using 12-bit, 250 MSPS ADC. In order to synchronize the clocks between the boards an external
Figure 4.8: FPGA and MSDPD clock Architecture

clock generator of 30.72 MHz is applied to the sys in port of the MSDPD. Using this reference clock the board generates the necessary frequencies such as the local oscillator, sampling frequencies for ADC and DAC, network clock. The network clock is applied as a reference clock to the DDR3 memory which in turn is used to generate the clock for the actual logic implemented on the FPGA.

The DAC in the transmit path is programed with a sampling frequency of 983.04 MHz i.e. with an interpolation of 4. The signal is upconverted using an I/Q modulator and sent to a 10W NXP PA. The frequency spacing between the bands is around 310 MHz where the lower band is transmitted at 1890 MHz and the upper band is
Figure 4.9: Comparison of the LSB and USB spectral performance of the 2D-LSCS, 2D-CS and 2D conventional polynomial models for the 2 test cases.

at 2200 MHz. A reference clock (refclk) as shown in Fig. 3.9 is used for synchronization between the boards. The output from the PA is filtered and fed back to the observation path on both the boards. The signal is downconverted to 184.32 MHz IF signal, digitized using 245.76MHz and stored in the FPGA’s DDR memory. The stored data can be used for further processing like digital down conversion (DDC), time alignment [15] using Matlab. An approximation of the inverse PA model can be extracted by using indirect learning as [17] and swapping the input and output in (4.24) to extract the coefficients for the DPD system.
The 2D-CS and 2D-LSCS and the 2D polynomial methods will be compared here. Both the 2D-CS and 2D-LSCS methods use a memory depth $ML$ of 3, and $N_{s1} = N_{s2} = 4$ spline intervals along the $|x_1|^2$ and $|x_2|^2$ directions. The 2D polynomial model relies on a memory depth of $ML$ of 3 and non-linearity order $NL$ of 7 in order to reach the best performance possible using this approach. The conventional 2D-CS case relies on the above 2D polynomial model for the initial gain identification.

Two test cases are used for evaluating the efficiency of the developed methods. In the first test case (Case I), a 3c-WCDMA signal with 3.84 MHz of bandwidth in each
carrier and LTE signal of 10 MHz are considered. In the second test case (Case II) a
3c-WCDMA signal with 3.84 MHz of bandwidth in each carrier is considered for both
the bands. A two step experimental procedure is used. In the first experimental step
(MATLAB), all the linearization algorithms are extracted and implemented on a host
computer and the predistorted signal data downloaded onto the DDR of the FPGA
testbed for transmission to the PA. In the second experimental step (FPGA), the
2D-LSCS and 2D polynomial architectures are implemented in the FPGA for the real
time linearization of the dual-band PA. The hybrid architecture is used to design both
the 2D-LSCS and 2D-polynomial on the FPGA with an LUT size of 256 for each 1D
basis function. As a result the 2D-LSCS and 2D-polynomial implementations exhibit
the same DPD latency (14 clocks in the example studied) since the hybrid architecture
is independent of the number and type of basis functions selected [38]. The extraction
of the basis coefficients are performed on the computer and only the updated basis
coefficients of the DPD model are downloaded onto the FPGA.

Using the testbed shown in Fig. 3.9, the performances of all the methods are
established for the same 10W PA for both test cases (I/II) and for both experimental
procedures (MATLAB/FPGA) by training the models on 8192 samples and testing
them on a longer frame with 65536 subsequent samples. For a fair comparison the
number of coefficients used in each model is selected such that each model achieves
its optimal performance.

The ACPR and NMSE performance results in each test case (I/II) and exper-
imental step (MATLAB/FPGA) with and without DPD, are tabulated in Table 4.2
for all methods. In both the test cases, the ACPR is calculated at a 5 MHz offset
from the signal; it is observed that the signals from the PA output do not fall below

68
Table 4.2: NMSE/ACPR for Dual-Bands with/without DPD.

<table>
<thead>
<tr>
<th></th>
<th>Test Case I</th>
<th></th>
<th>Test Case II</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3-WCDMA (LSB)</td>
<td>LFE(10MHz) (USB)</td>
<td>3-WCDMA (LSB)</td>
<td>LFE(10MHz) (USB)</td>
</tr>
<tr>
<td></td>
<td>NMSE (dB)</td>
<td>ACPR (dBc) @+/−5MHz</td>
<td>NMSE (dB)</td>
<td>ACPR (dBc) @+/−5MHz</td>
</tr>
<tr>
<td>No DPD</td>
<td>-23.70</td>
<td>-30.11/-30.61</td>
<td>-22.55</td>
<td>-30.69/-30.83</td>
</tr>
<tr>
<td>2D Conv. Poly</td>
<td>-40.68</td>
<td>-50.21/-49.78</td>
<td>-41.25</td>
<td>-49.11/-49.71</td>
</tr>
<tr>
<td>2D-CS</td>
<td>-39.19</td>
<td>-48.12/-48.03</td>
<td>-37.86</td>
<td>-48.65/-47.88</td>
</tr>
<tr>
<td>2D-LSCS</td>
<td>-43.07</td>
<td>-52.91/-52.75</td>
<td>-41.46</td>
<td>-52.98/-53.25</td>
</tr>
<tr>
<td>FPGA: 2D-Poly</td>
<td>-39.30</td>
<td>-49.80/-49.17</td>
<td>-38.21</td>
<td>-49.65/-49.51</td>
</tr>
<tr>
<td>FPGA: 2D-LSCS</td>
<td>-42.56</td>
<td>-52.54/-52.26</td>
<td>-41.59</td>
<td>-52.71/-53.26</td>
</tr>
</tbody>
</table>

An improvement of about 3 dB in both ACPR and NSME is observed when compared to other models.

The lower (LSB) and upper (USB) sideband spectra obtained with the new proposed 2D-LSCS (blue line & triangles), the conventional 2D-CS (green line & squares) and the reference 2D Polynomial (red line & circles) models for the two above test cases I and II, are compared in Fig. 4.9 (a),(b),(c),(d) when both the extraction and the linearization are performed on a computer and the data are downloaded onto the FPGA testbed to be transmitted to the PA. For reference the PA output without DPD is shown using black line. In Fig. 4.10 (a),(b),(c),(d) the performance of 2D-LSCS (blue line & rectangles) and 2D-Polynomial (red line & circles) are compared for the case where the predistortion linearization of the dual-band waveforms is performed

the mask requirement of -45dbc without DPD. As tabulated in Table 4.2, it can be observed that the best ACPR is obtained using the new proposed 2D-LSCS approach.
in real-time inside the FPGA using the hybrid architecture of Fig. 4.6. In both the LSB and USB spectra for the off-line MATLAB predistortion of Fig. 4.9 and for the real-time FPGA predistortion of Fig. 4.10, the 2D-LSCS is observed to provide the best performance with up to 3 dB reduction in sideband spectral density compared to the reference 2D polynomial model.

<table>
<thead>
<tr>
<th>2D Model</th>
<th>Number of Complex Coefficients</th>
<th>General</th>
<th>TestCases</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D Polynomial</td>
<td>$ML \cdot NL \cdot (NL + 1)$</td>
<td>168</td>
<td></td>
</tr>
<tr>
<td>2D-CS</td>
<td>$2 \cdot ML \cdot N_{s_1} \cdot N_{s_2} \cdot 16$</td>
<td>1536</td>
<td></td>
</tr>
<tr>
<td>2D-LSCS</td>
<td>$2 \cdot ML \cdot (N_{s_1} + 1) \cdot (N_{s_2} + 1)$</td>
<td>150</td>
<td></td>
</tr>
</tbody>
</table>

The number of coefficients required by each model to achieve its best performance is given in Table 4.3. It is observed that 2D-LSCS requires fewer coefficients, while achieving a better performance when compared to other models. The overall extraction time including both the matrix setup and its inversion is also shown in Table 4.4. In the present testbed the DPD extraction is performed in MATLAB. The number of floating point operations per second (FLOPS) used in the inverse of the matrix for the coefficient extraction is reduced [39] as shown in Table 4.4. The extraction of 2D-LSCS is thus substantially faster than that of the 2D polynomial case. This is due to the fact that 2D-LSCS relies on piece-wise polynomials of reduced order and requires a smaller number of basis functions.

The reduction of the logic resources used for the real-time implementation of the linearization algorithm in the FPGA is also important to yielding a low cost
Table 4.4: Comparison of the DPD extraction time and real-time DSP resource utilization.

<table>
<thead>
<tr>
<th>Basis Function</th>
<th>DPD Extraction</th>
<th>DSP Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time (sec)</td>
<td>FLOPS (Giga)</td>
</tr>
<tr>
<td>2D-Polynomial</td>
<td>0.62</td>
<td>9.2</td>
</tr>
<tr>
<td>2D-LSCS</td>
<td>0.5</td>
<td>8.2</td>
</tr>
</tbody>
</table>

implementation. It can be verified from Table 4.4 that the total number of real multipliers, adders and LUT memory used by the 2D-LSCS model is less than that used in the 2D Polynomial model in the efficient hybrid implementation of Fig. 4.6. The LUTs which are used to store the 1D basis function are synthesized in the FPGA using Stratix IV M9K blocks which can be instantiated as dual-port memories [40]. Since the same basis functions $\phi_{ij}$ are used for both the LSB and USB bands, the LUT memories benefit from being implemented using dual-inputs and dual-outputs. The multipliers and adders are used from the library of parametrized modules (LPM) available from Altera intellectual property (IP) suite. The LUT values are generated offline to initialize the M9K blocks and never updated.

Table 4.5: FPGA resource utilization

<table>
<thead>
<tr>
<th>Mode</th>
<th>Logic Utilization</th>
<th>ALUT</th>
<th>M9K Blocks</th>
<th>Registers</th>
<th>Signed Multipliers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polynomial</td>
<td>12511</td>
<td>6253</td>
<td>6</td>
<td>13666</td>
<td>392</td>
</tr>
<tr>
<td>LSCS</td>
<td>10948</td>
<td>5528</td>
<td>5</td>
<td>11469</td>
<td>378</td>
</tr>
</tbody>
</table>
The FPGA resources used by the 2D-LSCS and 2D Polynomials are tabulated in Table 4.5 for the sake of comparison. Most of the combinational logic is implemented inside an FPGA using adaptive look-up tables (ALUT). The Logic Utilization parameter provides a good representation of all the resources (ALUT, M9K memory blocks, Registers, Signed Multipliers) used inside an FPGA for the real-time calculation. The Logic Utilization parameter in Table 4.5 confirms that in the actual hardware implementation, 2D-LSCS utilizes less FPGA resources than the polynomial representation.
CHAPTER 5

Design Methodology

DPD algorithm development needs to come up with a suitable predistorter and adaptation model for a specific PA. The resulting DPD scheme needs to meet standard requirements, be resource efficient on target platform, and flexible to changes. For this to happen in minimum time to market, a joint software simulation and flexible hardware platform with following properties is proposed [41]

i) Enable to easily transmit and receive different waveforms including standard test patterns and custom carrier aggregation modes.

ii) Enable testing DPD in isolation from other radio head systems such as crest factor reduction.

iii) Facilitate performance benchmarking for various algorithms and enable swapping components of a DPD solution

iv) Provide design and hardware modularity

The different tools that are used to decrease the turnaround time for the development are discussed here.
Altera RF Framework

Altera RF Framework is used to support the development of Wireless RF-card applications and similar systems. It provides a means of integrating third-party mixed signal RF development boards like MSDPD with an Altera FPGA development board (Stratix IV) and interfacing these to Matlab as a host development and analysis environment [5]. The RF framework provides the necessary IPs to interact with the hardware using Matlab. The wireless RF framework provides support to the developers to evaluate DPD algorithms. The RF framework can provide the necessary real-time operation with the analog hardware and accurately evaluate the algorithm’s capability to adapt with the changes in the PA characteristics. The framework allows downloading waveforms to the FPGA and then play out through the transmit path of ADI’s MSDPD while capturing large contiguous blocks of data from the receive path. These blocks of data can be read back to matlab for some data analysis and perform signal processing as shown in Fig. 5.1. The hardware setup resembles of a real-time base-station downlink transmit path with a feedback DPD observation path. Both the transmit and receive blocks operate at the sampling frequency required by the DAC and ADC. DDR3 memory on the board is used for transmitting and receiving waveforms.

Qsys

Qsys is a system level integration tool which saves significant time and effort in the FPGA design process. Unlike the traditional way of writing HDL modules to manually wire together the subsystem, Qsys automatically generate interconnect logic to connect between intellectual properties (IP) and sub-systems. It also generates HDL
files defining all the components of the system and a top level HDL file connecting all the components together. Qsys has the capability of generating either VHDL or Verilog files. An example of the Qsys system used in this work is shown in Fig. 5.2.

**DSP Builder**

DSP Builder is a tool box developed by Altera as an integrated part of Matlab’s simulink to go from system simulation to system implementation in few minutes. This tool automatically generates time optimized register transfer level (RTL) code based on the design in simulink. All the simulink blocks from dsp builder are available in advanced and standard blockset which are used to develop signal processing algorithms. DSP-builder is a schematic tool where you can visualize the components similar to simulink and they can be used simply by drag and drop onto the simulink
workspace. The design made in the dsp-builder maps to an optimized design onto the FPGA. The simulations performed on the design developed in DSP builder is bit and cycle accurate. The advantage of using dsp-builder over using VHDL or schematic editor is the compilation times. Since DPD designs are complex and big, the compilation time would be $\approx 15$ mins. It takes very long time to compile and simulate even a minor change and verify the effect. DSP builder can simulate the design without compilation thereby saving a lot of design time. The top-level DSP builder design of 2D LSCS is shown in Fig. 5.4. The functionality of different blocks that are shown in the Fig. 5.4 are given below:

i) Block 1 : This blocks has the dual-input memory blocks that store the 1D basis functions of the LSCS. The tensor product of these 1D basis is performed to generate the 2D basis function.

ii) Block 2,3 : The delay blocks used to generate the delayed input to include the memory effect.

iii) Block 4,5,6 : The actual implementation of the model where the coefficients are multiplied with the basis function.

iv) Block 7 : Computing the envelop square of the input.

v) Block 8 : Calculating the address of the memory location depending on the envelop square of the input.

The following design methodology is proposed and summarized on Fig.5.3. In the first step, the setup is used as a regular VSG/VSA solution to ensure the DPD and related algorithms meet the expected results and performances while benefiting
from the flexibility and simplicity of the MATLAB environment. At this stage, the predistorter system is implemented in MATLAB, and both baseband predistorted signals are written onto the FPGA memory to be played. Then the received signal is analyzed and the DPD coefficients are recursively optimized by writing the new set of predistorted data to the DDR3 memory.

In the second step, once the DPD system meets the specifications, the FPGA implementation is started by using a bit/cycle accurate model of the signal processing components using DSP builder [42]. DSP Builder includes basic FPGA block models such as adders, multipliers or delays, and enables to build and verify the user’s digital system with real hardware parameters without requiring FPGA implementation in the system. Once the DPD model is built, the generated outputs are tested by transferring them to the FPGA memory and playing them to ensure that they provide similar
performance to the MATLAB implementation. Moreover, these high level synthesis tools are able to translate to RTL code directly via an automated process.

Finally, the generated RTL code is integrated in the overall communication system using standard interfaces and commercial integration tools like quartus and qsys [43] [44]. By the end of this process, an image file is generated by the compiler for a target FPGA and then downloaded to the target FPGA device. At this stage, only the signal processing and the coefficient estimation is done in MATLAB which can furthermore be implemented on a different dedicated signal processor.
Figure 5.4: DPD design using DSP Builder

Figure 5.5: DPD design using Quartus
CHAPTER 6

Conclusion

A 2D multisine signal is used to excite PA for system identification of 2 band PAs. It is applied to a GaN PA and observed that the normalized PA output saturates at approximately the average output envelope power $|y_1|^2 + |y_2|^2$. This possibly originates from the thermal response of the GaN HEMT which can follow the slower average envelope power $|y_1|^2 + |y_2|^2$ but not the faster peak envelope $|y_1| + |y_2|$, which is modulated by the band beating (a 310 MHz offset is used here between the lower and upper bands).

A new single band direct learning technique referred as Quasi Exact Inverse is designed which extracts the PD coefficients directly from the PA model. Measurement results show that the developed algorithm performs better linearization when compared to in-direct learning architecture. Even though same number of coefficients are being used in both the models, still there is improvement of upto 2dB in terms of ACPR and NMSE. With the improvement in single band system, the QEI is also extended to dual-band system. A new dual-band 2D quasi-exact inverse for PA model using direct learning architecture and memory polynomials is developed. The performance is compared with an indirect learning architecture using 2D MP. It is observed that when using the same number of coefficients, there is a 4 dB improvement in both
NMSE and ACPR in the presence of the additive uniform white noise. The improvement is obtained without any iteration as in the case of real-time system where the data is never played twice. Since the performance of QEI depends on the accuracy of the PA model, better linearity can be achieved for more accurate PA models.

To relish the advantages of piece-wise polynomials, cubic splines are developed for both single and dual band system. A new single band LSCS is developed using indirect learning architecture. It is observed that a 2 dB improvement in NMSE and ACPR is observed when the PA is excited with a 3c-WCDMA signal. The dual-band DPD system is also developed and implemented using two different C-spline basis and their performance are compared with each other and also with a traditional 2D polynomial model. The method referred to as 2D-LSCS, has the advantage over the 2D-CS method, in that it can be directly extracted from the measured data using the least square method without depending on any other model. Besides retaining the non-oscillatory advantage of a low-order piece-polynomial representation, the new 2D C-spline basis also exhibits a local-focus (2D Kronecker delta property) and relies on a non-uniform knot distribution to better account for the data distribution. All these attributes are engineered to increase the extraction robustness. The new LSCS approach can be implemented using a hybrid architecture [38] which stores the 1D basis functions in LUTs and numerically computes the 2D basis functions using tensor product multipliers. This architecture is selected to optimize the management of the FPGA DSP resources and reduce the real-time latency. Both LSCS and 2D polynomials uses the same implementation for a fair comparison in the resource usage.

These DPDs can be easily implemented on an FPGA or application specific integrated circuits that are used to linearize the PA. The new 2D-LSCS achieved the
best performance with fewer FPGA resources, fewer basis coefficients and reduced extraction time, when compared to 2D C-spline and 2D polynomial models for the two dual-band scenarios studied. The decrease in the number of basis coefficients and basis functions reduces the computational resource utilization of the FPGA. Furthermore the extraction time is also substantially reduced due to the reduced number of coefficients and the reduced polynomial order. The measurement results show that even though 2D-LSCS utilizes less real-time DSP/FPGA resources and requires less extraction time, a performance improvement of upto 3 dB in NMSE and ACPR is still observed compared to the conventional 2D polynomial model. The improvement in performance of the 2D-LSCS method over the 2D polynomial is observed not during training but in the linearization of subsequent data for which the dual band linearization system was not trained for. This increased performance for the new 2D-LSCS extraction originates from the engineered characteristics of the new proposed 2D C-spline basis.
CHAPTER 7

Future Work

7.1 Full Transmitter

7.1.1 Crest Factor Reduction

The high peak to average power ratio (PAPR) of the signal causes In-band and Out-Of-band distortion. The crest factor reduction (CFR) is a digital technique used to reduce the PAPR of the signal so that the amplifier can be operated more closer to the saturation point, thereby increasing the efficiency of PA and also allowing higher average output power to be transmitted before saturation occurs. CFR is often incorporated with DPD in the transmit chain to linearize the PA. CFR eases the linearization by reducing the dynamic range of the signal. There are many CFR techniques that are currently being implemented for single band [45] [46], but the performance degrades when they are used for multiband because the peak should be considered for the composite signal. When a 1D CFR is used in a dual band case, it either perform over-reduction (more than necessary) or under-reduction (not enough). Hence a 2D CFR technique is developed for taking into account both the bands in [47] [48]. The sum $P_{sum}$ or a combination of both the bands is utilized to compare with a threshold value instead of comparing the individual power separately.
A more compact DPD algorithm can be developed by offsetting the coefficients of the PD model to achieve a low PAPR predistorted dual-band signal, as developed for a single band case in [49].

7.1.2 Digital Up Converter (DUC)

Currently the DUC is performed inside the ADi’s MSDPD board and is limited to only 1 value i.e. 184.32 MHz. The design can enable a variable value for IF by implementing this component inside the FPGA. This feature will include the flexibility of changing the frequency of operation which is dominant feature requirement for SDR architecture.

7.2 Full Receiver

7.2.1 Digital Down Converter (DDC)

In the current setup DDC is also performed inside the ADi’s MSDPD board and is limited to only 1 value i.e. 184.32 MHz. If this frequency can be designed as tunable, then the setup can be made more close to SDR system.

7.2.2 Time-alignment

This is the most critical block which synchronizes the amplified data with the input for coefficient extraction. The quality of extraction depends on how well the output is synchronized with the input. This alignment can be done inside an FPGA by interpolating, delaying and then decimating back to the original sampling frequency.
7.2.3 Matrix Inversion

The coefficients of a PD model are extracted by finding the inverse of a matrix constructed from basis function and input. A 64x64 matrix inversion is readily available in Stratix IV FPGA. This block can be used for coefficient extraction inside an FPGA.

By implementing all these components inside the FPGA, it will enable to explore more complex algorithms like NARMA modeling, real-time adaptation and recursive least square extractor.

7.3 New Algorithms

7.3.1 Dual Feedback Architecture

A new multi-stage DPD based on ILA is developed in [50]. The DPD is implemented in two or three stages which reduces the computational complexity for PD extraction. Two algorithms are developed using a single feedback loop as follows:

1. Algorithm 1: A system level identification is performed for each stage. Once the extraction process converges to the best possible solution a new stage is added. A number of stages are added until the solution converge.

2. Algorithm 2: In this case all the stages are considered at the same time. The system is optimized taking all the stages into consideration.

A dual loop parameter characterization structure is used in [51] to improve the accuracy of the PD extraction. In conjunction with the inverse model structure, a reference model structure is also used for fine tuning the coefficients. In the reference
model structure, the coefficients are optimized by comparing the output of PA and the input of PD. There is only one feedback loop for the entire PA chain.

A new feedback scheme has to be studied where the output from all the PAs are observed and a DPD model for each PA is developed and implemented in the transmit chain. The feedback scheme should be tested with single band and can be later extended to dual-band system.

The linearization of each PA can be performed with a dedicated DPD instead of single DPD for the entire chain as shown in Fig. 7.1. The single PA shown in the Fig. 7.1 can also be a chain but since two DPDs are used, the complexity of the DPD model can be reduced. The power amplifier model (PAM) for each PA are extracted from the dual feedback structure and a QEI model is derived to extract the coefficients for both the DPDs.

A B-spline model is used for the gain function in (3.10) to model the PA. The different model parameters that are used in this simulation are $M_A = 2, M_P = 5$. It is observed that an NMSE of -120dB is obtained when compared to -54dB, using single
stage DPD. It can be verified from Fig. 7.2(b) that the new architecture improves the linearization when compared to single DPD. This scheme should be evaluated on a real amplifier and testbed to verify the improvement in the performance in terms of NMSE and ACPR.

### 7.3.2 Reduced Complexity for Multiband DPD

The general memory model used to represent the output of PA for each band is shown in (1.3). As the number of bands that are concurrently amplified by a PA increases, the complexity of the model also increases. The number of coefficients required are exponentially increased with the number of bands [8] [52].

The complexity of multi-band system is greatly reduced in [53] by using a single feedback loop DPD architecture and different band-limited functions can be added into a single DPD model to separate the frequency components for each dedicated...
band. The number of coefficients are greatly reduced since only single set of coefficients are used. This architecture requires only 1 feedback since the data sequence no longer requires to capture at the same time. Since all the carriers are aggregated the required sampling frequency will be high when compared to frequency selective technique.

A new lower complexity DPD model should be designed which can be extended to higher number of bands. For example a 2D model can be developed to model a 3 band signal by taking into consideration either $P_{avg}$, $P_{max}$ or $P_{min}$ of the composite signal. Further investigation must be performed in this aspect to reduce the complexity for multi-band systems.

7.3.3 Adaptation

The robustness of any model can by verified by extracting the DPD coefficients from a shorter frame and implementing it on subsequent longer frame. An iterative process is used to extract the coefficients from the shorter frame so that they are optimized and valid for longer frame. In the current iterative process the same frame is being used for all the iterations, hence when the longer frame is used with different average power or peak power the performance degrades. A weighted algorithm should be developed to assign weights to the coefficients extracted from different frames selected along the entire longer frame.
BIBLIOGRAPHY


