RICE UNIVERSITY

Nonlinearity Correction in Massive MIMO Systems via Virtual DPD

By

Chance Tarver

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

Doctor of Philosophy

APPROVED, THESIS COMMITTEE

Joseph Cavallaro

Joseph Cavallaro

Professor of Electrical and Computer Engineering and Computer Science

Christoph Studer

Christoph Studer (Aug 8, 2022 16:36 GMT+2)

Christopher Studer

Associate Professor of Integrated Information Processing, ETH Zurich

Ashutosh Sabharwal

Ernest Dell Butcher Professor of Electrical and Computer Engineering

Alexios Balatsoukas Stimming (Aug 9, 2022 17:01 GMT+2)

Alexios Balatsoukas Stimming

Assistant Professor of Electrical Engineering, Eindhoven University of Technology

Αναστασιος Κυριλλιδης

Αναστασιος Κυριλλιδης (Aug 9, 2022 08:43 GMT+3)

Anastasios Kyrillidis

Noah Harding Assistant Professor, Computer Science

HOUSTON, TEXAS

August 2022

Abstract

Nonlinearity Correction in Massive MIMO Systems via Virtual DPD

by

Chance A. Tarver

Many antenna massive multiple-input, multiple-output (MIMO) arrays are a key technology in 5G and beyond. Practical deployments include nonlinear power amplifiers (PAs) to amplify the transmitted signals and overcome path loss in the channel. However, these nonlinearities degrade the user error vector magnitude (EVM) and cause out-of-band (OOB) emissions that harm the signal-to-noise ratio (SNR) of users of adjacent channels. In legacy single antenna and MIMO systems with a low number of antennas, this is solved by learning an inverse model of the PAs and performing digital predistortion (DPD) before each PA. As the number of antennas in the array grows, the computational burden of the DPD grows significantly. Moreover, the exact nature of the nonlinearities is not straightforward in massive MIMO scenarios that involve beamforming, potentially with multiple users.

In this work, we seek to answer many fundamental questions about the effects of nonlinear PAs in massive MIMO. We collect and present measurements from Doherty PAs in a 16T MIMO array with 491.52 MHz of capture bandwidth at 3.5 GHz and 100 MHz 5G new radio (NR) signals as well as beamformed results from the Reconfigurable Eco-system for Next-generation End-to-end Wireless (RENEW) basestation. We use the

measurements to assist in developing mathematical models and simulators. We find that for a single-user system, the OOB energy is dominant in the direction of the main beam. Moreover, in multi-user scenarios, distinct spatial intermodulation beams of OOB energy appear in unique directions distinct from the intended users. These spurious beams may potentially desensitize victim users of adjacent channels. We create a virtual DPD (vDPD) solution that moves the DPD block to predistort before the massive MIMO precoder where the dimensionality is lower, reducing complexity in some scenarios. Our novel vDPD scheme uses a neural network (NN) to learn the function to linearize the effective nonlinearity in each beam.

Acknowledgements

Many thanks to all the people who helped me complete this Ph.D. Many people throughout my life have helped set me on this path. These people from my childhood, grade school, undergrad, and beyond have had a tremendous impact. Below I briefly give acknowledgments to a few of the many that deserve thanks.

My wife, Jennifer, deserves more gratitude than can fit in these pages. Her continuous patience, understanding, and encouragement through all the nights and weekends I worked are a true sign of devotion and love. Thanks, Jenny, for all of your love and support. I couldn't do this without you!

It is not an overstatement to say I would not be here without the support of my family. My parents prioritized my siblings and me in every possible way, and I am thankful for the opportunities their sacrifices have created for me. To my mom, thanks for always being available to chat. To my dad, thanks for teaching me to work hard at everything. Grandpa, thanks for always being interested and supportive. Jessica and Jason, thanks for being there. Jim and Wendy, thanks for your support and always letting me stay at your house when I needed to visit Houston.

I finished this thesis remotely while working at Samsung. I am grateful to my Samsung colleagues, who provided support and technical feedback throughout this time. The skills I learned while working at Samsung made a huge difference in my work. Thanks, Gary, Shadi, and Khurram, for the guidance and career support as I worked on finishing the Ph.D. remotely. Thanks to my work team, Shunyao, Matt, Yu, and Sandy. Special thanks to Matt for listening to my Ph.D. progress every morning over a burrito.

Thanks to my Ph.D. committee. Thanks, Dr. Sabharwal and Dr. Kyrillidis, for being willing to join my committee and providing helpful feedback and evaluation on this work. Special thanks to Christoph and Alexios for meeting with me weekly over the last few years. Your continuous technical feedback made this work possible.

Finally, thanks to my advisor, Dr. Joseph Cavallaro. Without his guidance and patience, none of this work would be possible. Working with him has created many opportunities in my life, and I am very grateful for all he has done.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Next-generation wireless systems should prioritize energy. This endeavor is part of a global effort to reduce energy consumption to combat climate change. While the current cellular standards prioritize enhanced mobile broadband (eMBB), ultra-reliable, low-latency communications (URLLC), and massive machine-type communications (mMTC), future versions of the wireless standards should include a target for energy. While no one can optimize the full wireless stack, there is often room for improvement at each layer. In this thesis, we consider reducing the digital predistortion (DPD) complexity.

Currently, operators are deploying 5G worldwide, and this will enable connection between more than a billion people and even more devices. Beyond the simple voice communication seen in the early cellular standards, 5G and future beyond 5GS (B5G) standards enable high throughput, low latency, and high connection density. These areas will enable new use cases such as augmented reality, remote surgery, self-driving cars, smart factories, and other areas of innovation. One key technology enabling the future of wireless is massive multiple-input, multiple-output (MIMO), where many

antennas simultaneously transmit to multiple users on the same frequency channel [1]. Multi-user (MU)-precoding enables this capacity gain and can save energy as extra energy is not wasted being transmitted in directions without users.

While massive MIMO can save energy and improve the radio access network (RAN), the communications literature often overlooks one critical component that threatens these goals — the power amplifier (PA). A modern PA is typically a solid-state device that amplifies some radio frequency (RF) signal to a higher power to overcome path loss between two radios. These devices typically consume most of the power budget of a base station (BS) [2], [3] and often operate at power-added efficiency (PAE) of less than 50%. Hence, as we scale up the number of PAs in a massive MIMO BS, controlling the power demands of the PAs becomes critical.

The PA can be operated near its saturation point to improve the PAE. For example, many PAs will have efficiencies near 20% when operating with "back off" in their linear region. While in saturation, many PAs may have energy efficiency as high as 60% [4]. However, the device becomes highly nonlinear when operating in saturation. The tradeoff between nonlinearity and efficiency is especially true for Doherty PAs, which are being considered for many 5G applications (including massive MIMO) due to their efficiency. These PAs present extreme linearity challenges due to their unique architecture containing dedicated carrier and peaking amplifiers [4]. The nonlinearity creates spectral regrowth around the signal carrier, leaking energy from our licensed spectrum into adjacent bands. This out-of-band (OOB) leakage energy could harm a user of an adjacent channel, degrading their signal-to-noise ratio (SNR). Standards and regulatory governing bodies such as 3rd Generation Partnership Project (3GPP) and the Federal Communications Commission (FCC) create spectral emission masks to prevent this by limiting the maximum energy an RF transmitter may leak. In the case of 5G in the sub-6 GHz spectrum, this limit is -45 dBc, or 45 dB below the main

carrier

5G and other radio access technologies (RATs) are not particularly helpful. 4G, 5G, and Wi-Fi physical layers (PHYs) utilize orthogonal frequency-division multiplexing (OFDM) due to its high spectral efficiency, low-complexity equalization, and simple matrix operations for other digital signal processing (DSP) such as massive MIMO precoding. However, multicarrier waveforms have a high peak-to-average power ratio (PAPR). Hence, the root-mean-square (RMS) signal power is already backed off from the saturation point of the PA so that the signal's peaks are within the operating range of the PA. Moreover, bandwidths as wide as 100 MHz are used in sub-6 GHz 5G, leading to more memory effects to account for in DPD processing and hence more complexity.

Most wireless systems currently use DPD and other techniques such as crest factor reduction (CFR) to reduce OOB emissions, improve in-band error vector magnitude (EVM), and increase PA efficiency [3], [5], [6]. A trained DPD module creates an inverse model of a PA so that the cascade of the DPD and the PA is linear. However, DPD is often computationally complex, requiring many Giga floating-point operations per second (GFLOPS) for a typical 5G transmitter. While this is tolerable in single antenna systems, it becomes a sizable computational burden as the number of antennas grows.

Single-input, single-output (SISO) DPD has been deployed for over 20 years and is well studied. However, massive MIMO introduces new challenges. In particular, due to beamforming, it is not necessarily obvious where the OOB spectral regrowth goes. The adjacent channel energy could potentially be coherent or noncoherent in any direction. While the straightforward approach may be to linearize each PA individually, this would significantly increase the total complexity of the DPD per base station. Hence, we explore methods to reduce the predistortion complexity for

MIMO systems in this work.

A few existing works have examined this problem. However, many of these works neglect vital aspects. For example, some combination of memory effects, PA variations, wide bandwidths, OFDM signals, and multi-user communications are frequently ignored, even though they are critical to modern 5G systems. Moreover, many works do not consider the effect of nonlinearity on victim receivers in adjacent bands and instead focus on improving their own user EVM. We seek to provide a fuller investigation that considers each of these in this work.

1.2 Contributions

This thesis provides a detailed analysis of the problem, existing solutions, and state-of-the-art. Then based on the findings, we introduce a novel solution that reduces the total running predistortion complexity for some scenarios. In particular, we perform the predistortion operation before the precoder so that predistortion is performed per beam instead of per antenna, linearizing the total nonlinear distortion in the far-field of the array.

1.3 Published Works

We have investigated various aspects of DPD and published in the area for the last five years. These are shown below along with their citation number from the bibliography.

[7] M. Abdelaziz, C. Tarver, K. Li, *et al.*, "Sub-band digital predistortion for noncontiguous transmissions: Algorithm development and real-time prototype implementation," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, 2015, pp. 1180–1186. DOI: 10.1109/ACSSC.2015.7421326

[8] M. Abdelaziz, L. Anttila, C. Tarver, *et al.*, "Low-complexity subband digital predistortion for spurious emission suppression in noncontiguous spectrum ac-

cess," *IEEE Transactions on Microwave Theory and Techniques*, vol. 64, no. 11, pp. 3501–3517, 2016. DOI: 10.1109/TMTT.2016.2602208

[9] C. Tarver, M. Abdelaziz, L. Anttila, *et al.*, "Low-complexity, sub-band DPD with sequential learning: Novel algorithms and WARPLab implementation," in *2016 IEEE International Workshop on Signal Processing Systems (SiPS)*, 2016, pp. 303–308. DOI: 10.1109/SiPS.2016.60

[10] C. Tarver, M. Abdelaziz, L. Anttila, *et al.*, "Multi component carrier, sub-band DPD and GNURadio implementation," in *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2017, pp. 1–4. DOI: 10.1109/ISCAS.2017.8050455

[11] K. Li, A. Ghazi, C. Tarver, *et al.*, "Parallel digital predistortion design on mobile GPU and embedded multicore CPU for mobile transmitters," *J. Signal Process. Syst.*, vol. 89, no. 3, pp. 417–430, 2017

[12] C. Tarver, A. Balatsoukas-Stimming, and J. R. Cavallaro, "Design and implementation of a neural network based predistorter for enhanced mobile broadband," in *2019 IEEE International Workshop on Signal Processing Systems (SiPS)*, 2019, pp. 296–301. DOI: 10.1109/SiPS47522.2019.9020606

[13] C. Tarver, L. Jiang, A. Sefidi, *et al.*, "Neural network DPD via backpropagation through a neural network model of the PA," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 358–362. DOI: 10.1109/IEEECONF44664.2019.9048910

[14] C. Tarver, A. Balatsoukas-Stimming, and J. R. Cavallaro, "Predistortion of OFDM waveforms using guard-band subcarriers," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*, 2020, pp. 12–16. DOI: 10.1109/IEEECONF51394.2020.9443468

[15] C. Tarver, A. Singhal, and J. R. Cavallaro, "GPU-based linearization of MIMO arrays," in *2020 IEEE Workshop on Signal Processing Systems (SiPS)*, 2020, pp. 1–5. DOI: 10.1109/SiPS50750.2020.9195239

[16] C. Tarver, A. Balatsoukas-Stimming, C. Studer, *et al.*, "OFDM-based beam-oriented digital predistortion for massive MIMO," in *IEEE Int. Sym. on Circuits and Systems*, 2021, pp. 1–5. DOI: 10.1109/ISCAS51556.2021.9401479

[17] C. Tarver, M. Tonnemacher, H. Chen, *et al.*, "GPU-based, LDPC decoding for 5G and beyond," *IEEE Open Journal of Circuits and Systems*, vol. 2, pp. 278–290, 2021. DOI: 10.1109/OJCAS.2020.3042448

[18] K. Li, J. McNaney, C. Tarver, *et al.*, "Design trade-offs for decentralized baseband processing in massive MU-MIMO systems," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 906–912. DOI: 10.1109/IEEECONF44664.2019.9048727

[19] H. Ji, Y. Kim, K. Muhammad, *et al.*, "Extending 5G TDD coverage with XDD: Cross division duplex," *IEEE Access*, vol. 9, pp. 51380–51392, 2021. DOI: 10.1109/ACCESS.2021.3068977

[20] C. Tarver, M. Tonnemacher, V. Chandrasekhar, *et al.*, "Enabling a "Use-or-Share" Framework for PAL–GAA Sharing in CBRS Networks via Reinforcement Learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 716–729, 2019. DOI: 10.1109/TCCN.2019.2929147

[21] M. Tonnemacher, C. Tarver, J. Cavallar, *et al.*, "Machine learning enhanced channel selection for unlicensed LTE," in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–10. DOI: 10.1109/DySPAN.2019.8935859

[22] C. Tarver, A. Balalsoukas-Slimining, C. Studer, *et al.*, "Virtual DPD neural network predistortion for OFDM-based MU-Massive MIMO," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, 2021, pp. 376–380. DOI: 10.1109/IEEECONF53345.2021.9723343

In [7]–[10], we focused on the effects of carrier aggregation creating harmful intermodulation products. This work has become valuable in the extension to MIMO in that the effect of sending multiple beams in different directions is similar to sending in separate channels, and many of the established concepts can carry over to the MIMO case. In [11], we implemented various DPD schemes on a mobile graphics processing unit (GPU). In [12], [13] we consider neural network (NN)-based distortion. We then later use the NN to predistort OFDM guard-band subcarriers [14]. In [15], we implement a memory polynomial (MP) DPD per antenna on a GPU for MIMO predistortion. In [16], we scale up the OFDM-based DPD processing to operate on multiple antennas. In [22], we perform a version of virtual DPD (vDPD) to operate on MU-MIMO.

I have also developed breadth throughout communication systems, as shown in the following published works, which are not necessarily DPD-related. In [17], we develop a GPU-based scheme for decoding low-density parity-check (LDPC) in 5G new radio (NR). In [18], we consider decentralized-based processing for MU-MIMO systems. In [19], we develop a novel duplexing scheme called cross-division duplexing (XDD) to

enhance uplink coverage. In [20], [21], we develop spectrum sharing schemes for Citizens Broadband Radio Service (CBRS) and Unlicensed LTE. Additional publications based on the work in this thesis may also be published.

1.4 Thesis Outline

The remainder of this document is organized as follows. In the next chapter, we present a background on linearization for SISO and MIMO systems. To perform simulations and experiments we developed a software suite to simulate the nonlinearities in massive MIMO called MIMO Simulator with Amplifiers (MIMOSA). We present this software framework in Chapter 3. Then, we present various investigations on MIMO nonlinearity including mathematical models, simulations exploring the problem when it becomes mathematically intractable, and measurements from a MIMO array. Finally, we present an overview of the vDPD algorithm, which performs predistortion before a precoder using a NN before concluding in Chapter 6.

Chapter 2

Background

In this chapter, we present the background work for multiple-input, multiple-output (MIMO) digital predistortion (DPD), including the relevant mathematical models and prior works.

2.1 MIMO Communications

While multi-antenna MIMO communications have been around for decades [23], massive MIMO was introduced more recently. In 2010, Thomas Marzetta published the seminal work proposing to let the number of antennas grow to infinity [24]. The unlimited number of base station antennas leads to capacity gains for the network as the effects of noise and fast fading disappear, and multiple users can be served for the same time/frequency resource through spatial diversity. In the following subsection, we present the main mathematical model considered throughout this work.

2.1.1 OFDM MU-MIMO Waveform

We consider a fully digital, multi-user (MU) massive MIMO base station (BS) system with N power amplifiers (PAs), each connected to a single antenna. The BS

serves U single antenna users per time slot. Without loss of generality, we restrict the presentation below to one orthogonal frequency-division multiplexing (OFDM) symbol. A symbol of data to the users is represented by the vector $[\mathbf{s}]_w \in \mathcal{O}^U$, where w indexes the OFDM tones from 1 to W and $\mathcal{O}$ represents the set of complex-valued constellation points. Pulse shaping is applied by including guard-band subcarriers that are normally empty.

Linear precoding is applied separately to each OFDM tone, generating W vectors $\mathbf{u}_w \in \mathbb{C}^N$ with $\mathbf{u}_w = \mathbf{G}_w \mathbf{s}_w$, where each element represents a precoded subcarrier for one antenna. Here, $\mathbf{G}_w \in \mathbb{C}^{N \times U}$ is the precoding matrix such as zero-forcing (ZF) or maximum ratio transmission (MRT). Each vector is remapped to contain all the tones per antenna, $[\mathbf{u}_1, \ldots, \mathbf{u}_W] = [\mathbf{a}_1, \ldots, \mathbf{a}_N]^T$, where each $\mathbf{a}_n$ is a W-dimensional vector containing all tones for antenna port $n \in \{1, \ldots, N\}$. At this point, the data is converted from the frequency domain to the time domain via the inverse discrete Fourier transform (IDFT), which is typically calculated via an inverse fast Fourier transform (IFFT). The data is reorganized to be serial instead of parallel, and a cyclic prefix is added. In many systems, windowing is also applied between symbol boundaries to improve the spectral shaping [25]. We express this time-domain representation for each antenna as the vector $\mathbf{x}_n$. This vector is upconverted to radio frequency (RF) where it is transmitted through a PA with nonlinear function $f_n(\cdot)$. The time-domain data for each antenna is given as $\hat{\mathbf{x}}_n = f_n(\mathbf{x}_n)$, where each vector can equivalently expressed as a discrete-time signal, such as $x(k) = [\mathbf{x}_n]_k$. The frequency-domain equivalent is given as $\hat{\mathbf{u}}_n$.

In OFDM systems, the channel is usually modeled in the frequency-domain for each tone w as, $\mathbf{y}_w = \mathbf{H}_w \hat{\mathbf{u}}_w + \mathbf{n}_w$, where $\mathbf{y}_w$ denotes the received data, $\mathbf{H}_w$ is the $U \times N$ channel vector, and $\mathbf{n}_w$ is a $U \times 1$ Gaussian random noise term.

2.1.2 Testbeds

To bridge the gap between theory and reality, a few groups took early strides to develop massive MIMO platforms. Each represents a massive accomplishment as there are many engineering design challenges when developing testbeds of this scale. In 2012, The Argos platform was established at Rice University[26]. The platform considered array sizes of up to 96 antennas and provided insight into realistic channels [27], full-duplex [28], and many other concepts. There are three generations of this testbed, with the latest being actively tested and providing results in the Platform for Open Wireless Data-driven Experimental Research (POWDER)-Reconfigurable Eco-system for Next-generation End-to-end Wireless (RENEW) platform as a collaboration between Rice University and University of Utah [29], [30]. Similarly, Lund University developed the LuMaMi platform supporting up to 100 antennas and ten spatial streams for 20 MHz OFDM [31]. This platform was later updated to support real-time operation [32]. Currently, massive MIMO is a commercial reality [33] to various extents, with Samsung currently providing 5G new radio (NR) solutions with 64 transmitters supporting up to 8 downlink spatial streams and over 200 MHz of bandwidth [34].

While each of these platforms has provided tremendous value to the community, PA nonlinearity has not been a primary research focus and was likely not considered in their design. Hence, there are challenges when using the existing MIMO testbeds for exploring this topic. For example, these platforms do not provide feedback paths for PA observation and DPD learning. They also were not designed while considering the typical 3-5x upsampling rates used to make performing DPD or PA characterization viable for most wideband signals of interest. Hence, the maximum useful bandwidth that can be linearized is often fairly small.

2.2 Linearization of SISO Systems

PAs were first used in audio applications in the early 1900s. Many variations were quickly developed, including what would become known as the Doherty PA in 1936 [35]. While the technology has changed from vacuum tubes to solid-state, many core developments from this time are still widespread today. For example, the Doherty amplifier is still widely used for its energy efficiency and is an essential component in 5G deployments [4].

Soon after the adoption of the PA, the nonlinear effects such as spectral regrowth became a problem, and feedforward and feedback schemes became used in the analog domain to correct the nonlinearities Beginning in the 1980s, DPD schemes were first considered as digital modulation schemes were also being widely adopted.

For any digital predistortion system, two aspects can often be interchanged: the predistortion model and the learning method. In the following sections, we present an overview of the most widely used technologies for each. For additional surveys on the topics, see [5], [3], and [6].

2.2.1 Nonlinear Models

Saleh Model

The dominant PA model until relatively recently was the Saleh model [36]. This memoryless model describes a device's AM-AM, input amplitude to output amplitude, and AM-PM, input amplitude to output phase distortion, characteristics with only four parameters.

$$A(x(t)) = \frac{\alpha_A x(t)}{1 + \beta_A x^2(t)};\ \Phi(x(t)) = \frac{\alpha_\Phi x(t)}{1 + \beta_\Phi x^2(t)} \tag{2.1}$$

In Eq. (2.1), the PA output modulus is given as a function of the input signal's envelope, $x(t)$.

While the subsequent models in this section are currently used more frequently, the Saleh model is still utilized for simulating PAs [37]. This nonlinear model is not commonly used as a predistorter. However, it may be inverted in some cases to create a predistorter function [38], or an inverse may be calculated in the form of a look-up-table (LUT) [39].

Volterra-Series Based Nonlinear Models

For the predistortion models, Volterra series [40] and its simplified variants such as the memory polynomial (MP) [41] and generalized memory polynomial (GMP) [42] are frequently used. The MP was first introduced in [43] and expanded on in [41] and is given in Eq. (2.2)

$$\hat{x}(k) = \sum_{p=1}^{P} \sum_{m=0}^{M} \alpha_{p,m} x(k-m) \left|x(k-m)\right|^{p-1}. \tag{2.2}$$

This model was expanded into the GMP, shown in Eq. (2.3), which adds additional cross terms between the signal and its envelope. The more expressive nature of this formulation while remaining linear in terms of the parameters has led to widespread adoption:

$$\begin{aligned} y_{GMP}(n) = &\sum_{p\in P_a} \sum_{m\in M_a} a_{pm} x(n-m) \left|x(n-m)\right|^{p-1} + \\ &\sum_{p\in P_b} \sum_{m\in M_b} \sum_{l\in L_b} b_{pm} x(n-m) \left|x(n-m-l)\right|^{p-1} + \\ &\sum_{p\in P_c} \sum_{m\in M_c} \sum_{l\in L_c} c_{pm} x(n-m) \left|x(n-m+l)\right|^{p-1}. \end{aligned} \tag{2.3}$$

For these models, odd-order terms are typically dominant and the even-order can

be eliminated [42]. Other variations also exist that compensate for local oscillator (LO) leakage and in-phase and quadrature (I/Q) signal imbalance [44].

Neural Network DPDs An alternative to Volterra-series-based models is the neural network (NN) [13], [45], [46]. While most NN-based predistorters are MultiLayer Perceptron (MLP)-based, other architectures are being considered and may provide benefits for memory effects. For example, recurrent neural networks (RNNs) have been considered in [47], and [48] introduces the long short-term memory (LSTM) for DPD. More recently, convolutional neural networks (CNNs) have also been considered [49], [50].

In [13] and other works, the authors consider a multilayer feedforward NN with H hidden layers and N neurons in each hidden layer. M time-domain inputs are given to the network to account for memory effects in the PA. For each sample, the real and imaginary components enter the NN on separate neurons. Let g denote a nonlinear activation function, and let $\mathbf{W}_i$ and $\mathbf{b}_i$ denote the weights matrices and bias vectors corresponding to the ith layer in the NN. The output of the first hidden layer at time instant n is

$$\mathbf{h}_1(n) = g\left(\mathbf{W}_1\begin{bmatrix}\Re(x(n))\\ \Im(x(n))\\ \vdots \\ \Re(x(n-M+1))\\ \Im(x(n-M+1))\end{bmatrix} + \mathbf{b}_1\right). \tag{2.4}$$

The output of hidden layer $i \geq 2$ is

$$\mathbf{h}_i(n) = g\left(\mathbf{W}_i \mathbf{h}_{i-1}(n) + \mathbf{b}_i\right). \tag{2.5}$$

Finally, the output of the network after hidden layer H is

$$\hat{\mathbf{x}}(n) = \mathbf{W}_{H+1} \mathbf{h}_H + \mathbf{b}_{H+1}, \tag{2.6}$$

where the first and second elements of $\hat{\mathbf{x}}$ represent the real and imaginary part of the signal, respectively. Complexity remains low when considering a rectified linear unit (ReLU) activation function, which can be implemented with a simple multiplexer. To further reduce the computational burden, a designer could consider options such as pruning [51] and quantization [52].

Neural networks and other machine learning-based techniques can provide benefits in that no particular model is needed. With a traditional model, such as the GMP, only features explicitly included in the model may be accounted for. To account for high order nonlinearities, memory effect, cross-terms, I/Q imbalance, etc., the model quickly can become intractable. However, machine learning techniques follow the data and learn the features that best minimize a loss function. While high-order polynomials suffer from aliasing when using low sample rates, NNs do not necessarily have the same limitations as they are model-free.

Other areas of communications have realized the potential value of neural networks. For example, [53] turns the entire communication system into a machine learning problem, and [54] uses deep learning for MIMO detection. Recently, the concept of unrolling signal processing and using machine learning tools and techniques has gained traction for a wide variety of applications, including low-density parity-check (LDPC) error-correcting code decoding. For a complete overview of a

wide range of applications of deep unfolding, see [55].

2.2.2 Learning Methods

With any predistortion method, it is necessary to train the model to become an effective inverse. However, this is challenging because the PA is not given. Moreover, even if a perfect model of the PA existed, many nonlinear functions are not invertible to where an inverse model for predistortion could easily be calculated. Hence, creative methods are needed to train the predistorter.

Indirect Learning Architecture In general, it is possible to solve for a given nonlinear system if it is linear in the parameters and if the input and output are known. In such as case, the parameters that minimize the model error can be optimized via methods such as least-squares. However, that is not directly the case with DPD, as the ideal output of the predistorter to create the linearized PA output is unknown. In [56], the authors solve this problem by introducing the indirect learning architecture (ILA). Before this development, the idea of the ILA was introduced to train neural network control systems [57].

The ILA introduces the idea of a postdistorter, with the input being the PA output. The difference between the output of the predistorter and postdistorter is the error which can be used to update the model through least-squares. Since its introduction in [56], the ILA has been widely adopted. In [41], the ILA is adopted to train an MP predistorter. The GMP is introduced with ILA-based learning along with a Newton method scheme for updating the pre and post-distorters.

Iterative Learning Control A recent alternative to ILA in the literature is iterative learning control (ILC). First introduced in [58], ILC adopts a technique from control theory to learn the correct predistorted signal. Then, once the predistorted

signal is identified, a generic predistortion model can be solved for to map the original input signal to the predistorted signal. When compared to a direct and indirect learning approach for training a GMP, the authors in [58] report improved adjacent channel leakage ratio (ACLR) reduction via their ILC-DPD technique.

Decorrelation A decorrelation method was used in [8] to predistort in scenarios with non-contiguous carrier aggregation. The PA feedback was sampled at spurious emissions caused by the intermodulations between carriers. Then this feedback was correlated with basis functions so that the inverse could be injected in the digital baseband before the digital-to-analog converter (DAC). This decorrelating technique had the advantage of reducing the necessary sampling rate in carrier aggregation scenarios as each sub-band could independently be sampled at low rates.

Backpropagation Backpropagation can also be used for PA and DPD identification. While backpropagation is typically associated with NNs, it can be used on other differential functions. When using backpropagation, we take the gradient of a loss function with respect to a function's parameters and update the parameters based on their partial derivative [59]. In [60], the authors learn an MP via gradient descent to model the PA nonlinearities in a full duplex system. In [13], we present a DPD solution where we learn a NN model of a PA and use this to aid in the training of a NN DPD through backpropagation. While backpropagation is a powerful learning method, it also has high complexity. It often requires the aid of a graphics processing unit (GPU) to perform the training in a reasonable amount of time, which may not be suited for low-power deployments.

Direct Learning Direct learning is a poorly defined term that is often used to refer to any technique that is not one of the previously mentioned techniques. Some

techniques that are often referred to as direct learning include a pth-inverse, which approximates the inverse coefficients of a Volterra series [40], to other closed loop estimators [61]. While direct learning can avoid issues such as bias that appear in the ILA, convergence is usually LMS-based and slow.

SISO DPD Comparison In Section 2.2.2, we compare a few popular implementations. However, the comparison is of limited value since there is no common frame of reference. [45] and [41] include only simulations. The nonlinearity experienced heavily depends on the PA, its technology (laterally-diffused metal-oxide semiconductor (LDMOS), gallium nitride (GaN), etc.), its biasing, and the RF frontend architecture. The performance also depends on signal characteristics such as the peak-to-average power ratio (PAPR), bandwidth, and root-mean-square (RMS) power of the PA input signal. Finally, the measured ACLR value depends on the definition of channel bandwidth used. Hence, this table is not meant as a performance comparison. Instead, it provides a quick overview of the variety of schemes and their place in the DPD history. For works without a stated ACLR, we estimate it based on the figures.

2.2.3 Implementations

For each of the previous methods, the DPD computation needs to be performed at a sufficient rate to support the desired communication signal bandwidth. There are various implementations available throughout the literature. In [12], the authors implement and compare NN and MP predistorters on field-programmable gate array (FPGA) and find that the NN can achieve better performance for lower complexity. In [63], an MP predistorter is implemented on FPGA using precalculated tables to reduce the hardware computation requirements.

GPUs are attractive for implementation due to their high degree of parallelism

Table 2.1: Comparison of Common SISO DPDs

Reference	Year	Model	Learning Method	Bandwidth (MHz)	ACLR (dBc) Before/After
[39]	1983	Saleh	Direct	—	—
[62]	1990	Saleh	Direct	0.030	-30/-60
[45]	1993	MLP	Backprop	—	—
[56]	1997	Volterra	ILA	—	—
[43]	2001	MP	Direct	5	-42/-52.6
[41]	2004	MP	ILA	—	-45/-70
[42]	2006	GMP	ILA	15	-40/-57.6
[63]	2010	MP	—	3.84	-35/-60
[8][1]	2016	MP	Decor.	5[2]	-35.6/-68.3
[58]	2016	GMP	ILC	5	-32.4/-58.6
[13][1]	2019	MLP	Backprop	10	-40/-50

[1] Author of this thesis also coauthored this work.
[2] There were two, 5 MHz component carriers in this work. For ACLR, the IM3+ spur was measured.

and ease of programming compared to other high-performance computing devices like FPGAs. Hence, they have been used for DPD [11] [64]. GPUs are also considered in other areas of software-defined radio (SDR) physical layers, including for MIMO processing. For example, in [65], GPUs are used for detection and beamforming in an MU-MIMO base station. In [17], we used GPUs for LDPC decoding. By porting more functionality into GPUs, the benefits of the GPU can be further realized as the data can stay on the GPU longer, avoiding time-consuming memory transfers into and out of the device.

2.3 Linearization of MIMO Systems

While single-input, single-output (SISO) DPD systems are relatively well studied, as outlined in the previous section, MIMO systems are still currently being explored. In this section, we provide an overview of this topic.

2.3.1 Early Analysis

The exploration of the directionality of antenna arrays began in [66]. In this early work, active phase arrays for satellite communications were explored, and the author found that "nonlinearities form beams that, in general, radiate in directions different from the principal beam directions." This conclusion would later be rediscovered in the literature over 30 years later in the context of fully digital beamforming for massive MIMO.

Shortly after the publication of massive MIMO [24], many works began to explore practical aspects of scaling up the number of digital transceivers, including the effect of PA nonlinearities.

One popular technique for analyzing the effect of nonlinearities was to treat them as uncorrelated noise [67]–[69]. In such a scenario where the nonlinearities are truly uncorrelated, the authors find that the effects are expected to diminish as the size of the array increases. However, the premise of uncorrelated noise has been shown to be inaccurate [70]. For example, many works have shown that the out-of-band (OOB) distortion will see array gain in line-of-sight (LoS) scenarios in the direction of the user.

2.3.2 PAPR Reduction Methods

Due to the degrees of freedom available, multiple works have focused on controlling the PAPR of the MIMO output [71], [72]. While these works can reduce the PAPR of the signal at each antenna, that is often not sufficient to improve the ACLR. However, PAPR reduction is a vital step that is taken in most practical applications and should be deployed with a DPD method. Future research is needed to explore the joint application of PAPR reduction and predistortion in the context of massive MIMO.

2.3.3 mmWave

Many systems are targeted to the millimeter wave (mmWave) bands or frequency range 2 (FR2) in the 3rd Generation Partnership Project (3GPP) nomenclature. In mmWave systems, the target ACLR is relaxed to -25 dBc. This leakage can be tolerated due to the increased path loss in these bands leading to less intercell interference with users of adjacent bands. Many works targeting this band are hybrid systems [73], where a combination of digital beamforming, power splitting, and analog phased-array beamforming are used to drive a larger number of antennas. Hybrid systems may reduce the necessary digital system complexity at the expense of degrees of freedom.

2.3.4 Crosstalk

When discussing impairments in massive MIMO systems, crosstalk, where the signal from one antenna couples to an adjacent element or circuit, is often also considered in the literature.

Nonlinear Crosstalk

[74] was one of the first works to consider this and experimentally showed the change in the output error vector magnitude (EVM) for a single PA. While this represents an important early work in the space, there are a few caveats that have not been fully addressed in the subsequent literature. Firstly, the output EVM of an individual power amplifier does not indicate all there is to know about the performance of a MIMO system. For example, in the presence of beamforming, the crosstalk effects will not necessarily combine with the full array gain, leading to better far-field performance. Secondly, nonlinear crosstalk occurring at the input of the PA was considered up to a level of -15 dB. In this experiment, the authors artificially inject this extreme

crosstalk. However, this benchtop experiment does nothing to tell engineers what level of crosstalk to naturally expect in a deployment. Regardless, many later works took this original paper to mean that crosstalk could be as high as -15 dBc and designed complicated algorithms to compensate. In practice, nonlinear crosstalk levels should never be this high. If the crosstalk levels were to approach that level, the most prudent thing for the designer to do would be to add additional physical shielding between the elements.

In [75], a cross-over DPD was developed where the DPD for each antenna includes the other antenna inputs in the model. This original work was considered for a 2×2 MIMO system. The work shows that amounts above -30 dB crosstalk can contribute significantly to degradation in DPD performance. Similar to the authors' prior work, this was a benchtop experiment that explored the effects as the amount of crosstalk was swept in a controlled manner. Since then, many subsequent works [46], [76] have incorrectly assumed that -30 dB to -15 dB is a reasonable amount of crosstalk to expect in practical massive MIMO systems. However, in practical multiple antenna platforms, nonlinear crosstalk is often better than -60 dB, with SDR platforms such as the Xilinx ZCU111 FPGA board reporting a typical crosstalk level of -70 dB [77].

Beam-dependent Backwards Crosstalk

The second form of crosstalk often considered is backwards crosstalk. In this scenario, a reverse wave comes into the output port of the device, creating a load modulation that can change the PA model. In a MIMO system, the reverse wave could be caused by the transmitted signal from an adjacent antenna element. Hence, many recent works focus on beam-dependent load modulation where the over-the-air (OTA) coupling depends on the direction the output beams are steered. [78] creates a NN-based method to compensate for this.

While this effect is well studied, and there are multiple proposed solutions, these works neglects the fact that the problem can largely be solved by the circulator, which is already present in most time-division duplexing (TDD) massive MIMO systems. Circulators are used between the PA and the antenna so that the RX signal from each antenna element can be directed to the RX chain, while the TX signal can be directed to the same antenna. Circulators can often have isolations of over 20 dB on the forward and backwards waves, and hence would dramatically reduce the concern over beam-dependent load modulation.

2.3.5 Beam-oriented Methods

One popular technique for MIMO linearization, which we expand on in this thesis, is beam-oriented DPD. In BO-DPD, the far-field pattern is linearized instead of linearizing each individual PA. This method was first presented in [79] and is particularly useful for hybrid systems where it is impossible to directly linearize each PA.

2.3.6 Precoding Methods

Some recent methods exploit the degrees of freedom in massive MIMO to predistort. In [80], precoding is performed to cancel the third-order nonlinearities. However, this comes at the expense of beam-forming gain. Moreover, multi-user schemes are not explored.

2.3.7 Other MIMO DPD Schemes

In [81], the authors adapted the decorrelation technique from [8] to provide a reduction in complexity across an array. However, this method is targeted at time-domain processing and is performed per antenna. The main approach of [81] is to replace the computationally expensive GMP model with a lower complexity scheme. [82] develops

a two-box DPD scheme for millimeter-wave massive MIMO, which compensates for the differences between PAs and then linearizes the whole array with a common DPD. However, OFDM MIMO is not considered; instead, the technique is designed around time-domain phased array processing. The authors of [82] do not consider the OOB emissions in non-user directions.

2.3.8 Experimental Measurement Results

Massive MIMO hardware is not easily obtained or built. Hence, there is a shortage of experimental measurements in the community. However, some works have performed proof-of-concept (PoC) measurements on the effect of nonlinear PAs in MIMO. [82] developed a mmWave array to verify their two-box DPD approach with four transmitters and 40 MHz signal bandwidth. [78] uses a 200-MHz 5G NR OFDM waveform and a 64-element active antenna array to predistort using a NN at a 28 GHz carrier.

2.3.9 Implementations

Currently, few works in the literature implement a DPD for massive MIMO. In principal, any implementation from Section 2.2.3 could be reapplied and scaled up by the number of PAs in the array. However, this may become particularly challenging when the size of the DPD implementation scales beyond what can easily fit in an FPGA, application-specific integrated circuit (ASIC), and GPU. Many MIMO systems require some degree of decentralization. For example, [65] and [18] considers a fully decentralized solution, while other implementations may have a primary centralized FPGA for the primary processing, then multiple other FPGAs for handling some subset individual RF streams.

In [15], we implement a single GPU for performing an MP DPD on each antenna stream. We found that the throughput that a single GPU can support falls by a

factor of two, with each doubling in the number of antennas. While this work focused on how fast the GPU could perform the necessary computation, it did not consider the massive challenge of moving the I/Q data out of the processor to the DACs.

2.4 Current Issues

Massive MIMO poses many challenges that are difficult to overcome. In Table 2.2, we compare existing works that include some sort of simulation or experimental test. This table is meant to provide an overview of the current state of the field and available results. These findings are not directly comparable due to the many differences between each experimental platform that need to be controlled for. While each work contributes to the literature in some manner, when looking at the whole of the available works, there are some apparent gaps. In particular, many existing works fall into one or more of the following limitations.

Table 2.2: Comparison of MIMO DPD Solutions

Reference	Year	Number of Active Elements	Number of Users	Signal Type	Bandwidth (MHz)	Carrier Frequency (GHz)	PA Power (dBm)	ACLR (dBc). Before/After	Other Notes
[81]	2017	100	—	LTE-A	20	—	22	-31.8/-56.0	Simulation. DPD per antenna.
[79]	2018	4 (Hybrid with 2 digital)	1	LTE	10	3.5	—	-35.6/-48.9	Focuses only on user ACLR
[82]	2019	4	1	LTE-A	40	27	14	-36.0/-53.8	Focuses only on user ACLR
[78]	2020	64 (Hybrid with 1 digital)	1	NR	200	28	42	-25/-33	
[50]	2022	32	8	NR	50	—	30	—	Simulation. Focuses on in-band in the user direction via symbol error rate (SER).

1. **Single user.** Works do not consider the case of multi user.

2. **Neglects spurious beams.** Many of the existing works for MU-MIMO do not consider the OOB power in non user directions.

3. **Scales with the number of antennas.** Some works focus on reducing the complexity per antenna. However, this can still be problematic for scenarios with many antennas.

4. **PA model simplifications.**Neglects high order nonlinearities, memory effects, and PA variability.

5. **Limited bandwidth.** Most works do not consider cases with bandwidths on the order of 100 MHz, similar to what is deployed in 5G NR.

6. **No experimental results.** Many works rely on simulation without measuring MIMO hardware.

We seek to remedy as many of these issues with the state of the art in our following investigations, experiments, and algorithms.

Chapter 3

MIMO Software Systems

To perform our simulations and testing, we required the ability to simulate a full end-to-end multiple-input, multiple-output (MIMO) system, including power amplifier (PA) nonlinearities. As we began our explorations, we developed a software suite called MIMO Simulator with Amplifiers (MIMOSA) to explore various research questions in MATLAB. We then extended this simulation platform to support current state-of-the-art tensor libraries supported in python such as PyTorch by creating a python package called MIMOSA for Python (MIMOSApy). Using our Python package, we are able to explore novel machine learning solutions to various problems in wireless communications such as neural network (NN)-based digital predistortion (DPD) for MIMO, machine-learning aided precoding, and a variety of other topics that are being explored as candidates for and artifical intellgence (AI)-based 6G [83]. In this chapter, we present an overview of the software environments to support the investigations throughout the thesis.

3.1 MIMOSA

Many existing open source options for exploring MIMO systems perform limited simulations with the primary purpose of exploring metrics such as capacity. However, these simulations do not consider the PA nonlinearity. For this reason, we developed our own simulation platform to support the investigations for this thesis. In this section, we introduce MIMOSA, a general purpose MIMO simulation environment that includes impairments such as PA nonlinearity [84].

3.1.1 Motivation

Throughout many projects, it is common to implement a variety of quick functions to be able to test out core, novel ideas in research. While this is useful for a quick proof-of-concept (PoC), these testbenches often get stretched to do a wide variety of tests. Often, the initial assumptions of the original code are forgotten as we create a "spaghetti code" where we hack together various code fragments to test a new idea. As the research meanders, the codebase becomes more difficult to come along as we try to overextend the usefulness of the original code. These sort of codebases are often passed around and contorted in new ways for new projects, becoming an unusable mess that becomes impossible to debug. However, development and research does not need to be this way. By emphasizing good software-design principals from the beginning, we can develop a library of tools that ultimately are multi-purposed, easy to maintain, and ultimately time-saving.

The key approach taken in this project was to develop a library instead of just a script for testing an idea. This approach is the inverse of how many projects begin. We decompose our ideal experiment into core pieces, implement those pieces as modules that can operate independently and flexibility, then we build an application that simply uses these modules. Whenever we need to test new ideas, we simply connect

and configure the well-designed modules in new ways, rather than having to untangle and decouple the ideas from an initial experiment.

3.1.2 Software Architecture

We implement an object-oriented software library in MATLAB using the factory design pattern [85]. The core principle is modularity. We develop independent objects for each signal processing block in our experiments. For example, we create a "Precoder" superclass that defines the generic abstract structure of precoder. We then implement various subclasses such as zero-forcing (ZF) and maximum ratio transmission (MRT). Similarly, we create separate classes for a variety of modulators, channels, PAs, and other digital signal processing (DSP) blocks. Each specific implementation is designed to be standalone to where it is fully featured and flexible enough to be used in a variety of contexts.

However, simple modularity is not sufficient for creating a powerful, reusable library. To maximize the usefulness of the classes, we need to be able to connect any module to any other module. For example, there may be cases where the precoder class output is connected directly to a PA class input while in other scenarios there is an additional DPD class in between. To make it to where all modules can easily work with all others, we create a common interface. All modules exchange data in the form of the "Signal" class. A Signal class object will contain a matrix of the signal data at some point in the experiment as well as other useful context such as the current sample rate and domain. Each core module in MIMOSA implements a "use()" method that consumes a Signal and produces a Signal. Each core module also contains a "requiredSampleRate" and "requiredDomain." In all "use()" methods, we begin by calling the "matchThis(requiredDomain, requiredSampleRate)" method in Signal. This makes it so that all blocks can communicate with the developer not

needing to worry about the domain and sample rate throughout. For example, a precoder may output a Signal in the frequency domain at a sample rate of 30.72 Msps. We may then connect this to a DPD that is configured to require time domain data at 122.88 Msps. Instead of the developer worrying about verifying the data at each step in the signal processing pipeline, the DPD module in this example would see that the incoming data is in the wrong domain and at the wrong sample rate and would automatically perform the inverse fast Fourier transform (IFFT) and upsampling. Through the use of this common interface throughout the MIMOSA library, we can automate many of the transformations that can easily be overlooked and provide simple error-checking on functions throughout. This feature reduces the amount of bugs that are often introduced in "copy-paste" style development and allows for rapid development of the core concepts being evaluated.

To build an application, the developer builds a "dataflow" that aggregates various core modules. The dataflow "use()" method then passes data from block to block throughout the dataflow. For example, a Signal object may flow from modem to precoder to PAs to a Channel to a UE modem in a dataflow. To maintain flexibility, each core block may have multiple subclasses. At runtime, a param struct is passed into the factory method of the core modules to instantiate the requested subclass with some settings. For example, a param struct may dictate to the dataflow to configure the Precoder as a ZF and to configure the PA to use the GMP subclass with nonlinearity order seven. This structure allows the developer to rapidly test and compare various versions with no code changes. These concepts can be seen in Fig. 3.1, which shows the UML diagram for a subset of the available classes.

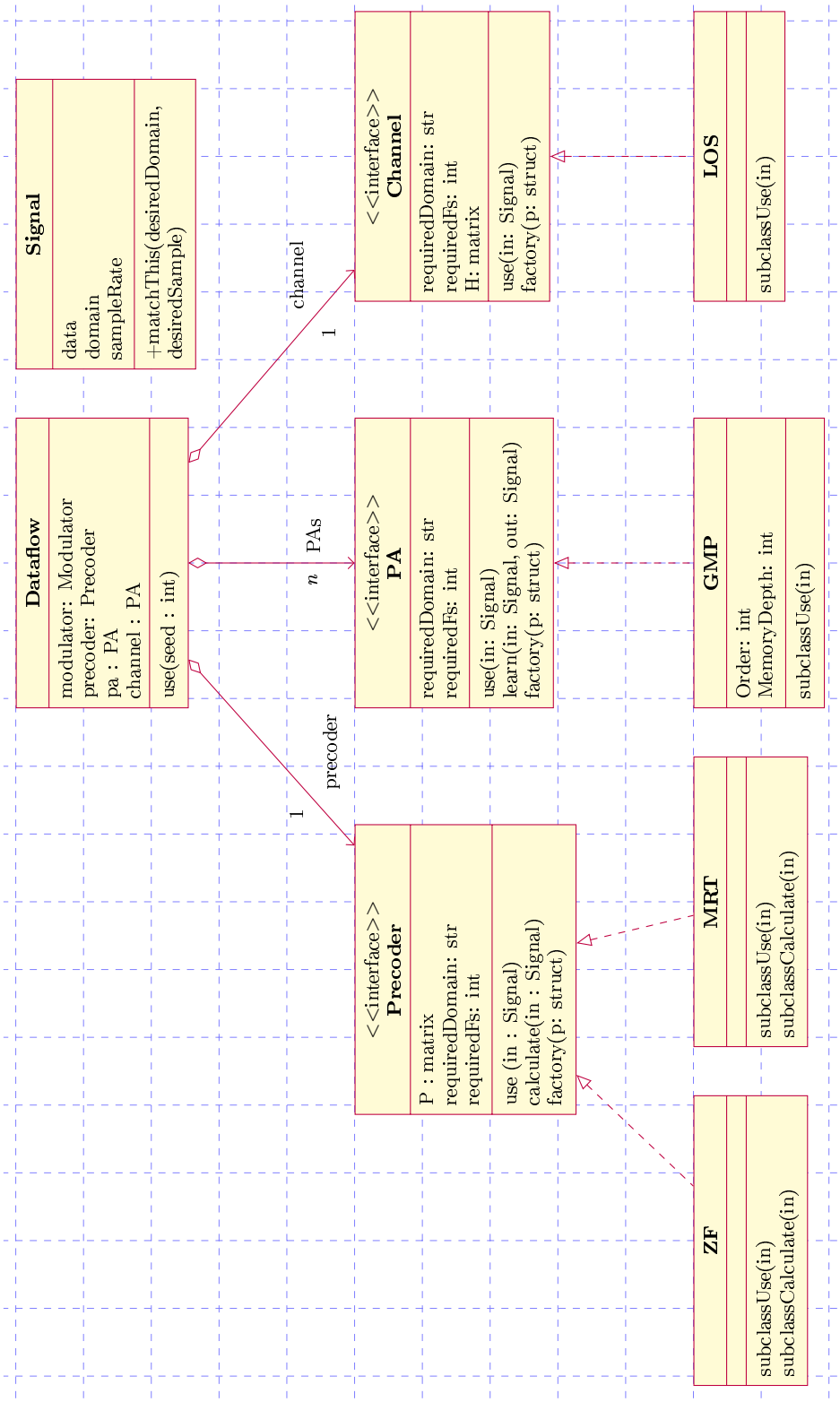


Figure 3.1: Subset of MIMOSA UML Diagram. The developer creates a dataflow for their experiment. This dataflow sets up various superclass blocks such as the Precoder and passes data from the use methods for each. The exact subclass implementation, such as ZF, is created in the superclass factory method at runtime.

3.1.3 Use Cases

MIMOSA is a flexible framework that can be used for a variety of simulation and experimentation needs. In this thesis, we use MIMOSA in Section 4.1.2 to explore PA variability. We also use the architecture to test the beamforming pattern using a combination of real PAs and a simulated channel in Section 4.1.3. While we focus on exploring the effect of nonlinearities in MIMO systems, the library could also be used to easily compare precoding strategies and other link-level simulator tasks.

3.2 MIMOSApy

While MIMOSA allowed for initial exploration of the problem of MIMO nonlinearities, the machine learning capabilities in MATLAB are somewhat limited. Throughout the community, most machine learning tasks are performed using libraries that are primarily for Python. For example, PyTorch is a powerful, general purpose tensor processing and machine learning package from Meta AI that is widely used in AI research. Hence, we then began to port much of the MIMOSA work to the Python environment in the form of a python package we call MIMOSApy.

3.2.1 Background and Design Options

Before beginning development of an extension to MIMOSA for machine learning, we considered multiple possible design options. The most critical choices include the language and the tensor processing backend.

MATLAB The first choice for building a system was MATLAB. The primary advantage provided by MATLAB is that it would allow us to build upon MIMOSA. However, MATLAB's machine learning packages are limited. The Rice Reconfigurable

Eco-system for Next-generation End-to-end Wireless (RENEW) project also provides an application programming interface (API) for MATLAB.

Keras/Tensorflow Keras is a popular interface library for python that provides an easy-to-use API for utilizing the Tensorflow machine learning engine [86]. While we initially tested many applications using Keras, we found that it was not well suited for custom applications and better for straightforward NN applications. Keras can utilize a graphics processing unit (GPU) for hardware acceleration, and we can effortlessly interface with RENEW using the Python libraries for RENEWLab.

PyTorch PyTorch is a machine learning library for python developed by Meta AI [87]. While often not considered as concise as the Keras library, it provides more control over the exact NN processing. In particular, PyTorch provides the ability to perform gradients automatically over any arbitrary functions written with the library. This ability allows us to easily perform optimization over an entire communication system with PA models, channel models, etc., whereas other packages require performing the neural-network training explicitly over neural networks. Because of this flexibility, we choose to develop on this framework.

3.2.2 Software Architecture

MIMOSApy is mostly a clone of the original MIMOSA software framework. We implement an object-oriented software library in python using the builder design pattern [85]. The architecture is similar to MIMOSA, with the exception of swapping the factory design pattern to the more general builder design pattern.

There is one critical difference throughout MIMOSApy when compared to MIMOSA. In our MATLAB implementation, all data uses a type of complex double. While the data for a precoder matrix or signal may be encapsulated in our of our

library classes, it is still using the MATLAB core type of a double. In our python implementation, we make all of our data represented as a PyTorch tensor. While these are usually single precision types, they allow us to leverage PyTorch tools anywhere in our system. The critical advantage of this system is that because everything is a tensor, the system automatically can keep track of gradients through all of our processing. For example, because the channel matrix is stored as a tensor, whenever we send data through a channel matrix, we can ultimately calculate some error metric and perform a backpropagation back through the channel and any other block.

3.2.3 Integration with RENEW

A key goal in this project is the ability to test with hardware. For testing with hardware systems, we developed a connection to RENEW. To accomplish this, we encapsulate the SoapySDR interface inside a subclass of the abstract interface Array class. This allows for a user to connect their work by simply building an array class and calling the TX/RX methods.

3.3 Conclusion

Software is a critical part of modern research. Many systems today are too mathematically intractable when considering the full complexity, such as high-order nonlinear power amplifiers operating on wideband systems. Hence, simulations are playing a larger role in research. Moreover, AI and machine-learning is playing a critical step in the future communications systems, which necessarily are implemented in software. Researchers today should take the time to develop good software engineering principles in their experiments to promote good code reuse, modularity, and repeatability. By developing MIMOSA and MIMOSApy, we hope to provide a platform that others

can use to bootstraps their own research onto while supporting these ideals.

Chapter 4

Initial Explorations

In this chapter, we explore the problem of power amplifier (PA) nonlinearity in massive multiple-input, multiple-output (MIMO) systems before providing algorithmic solutions in Chapter 5. We begin with measurements from a 16T MIMO platform to characterize wideband PAs. Then we collect beamforming measurements from the Rice Reconfigurable Eco-system for Next-generation End-to-end Wireless (RENEW) platform. Finally, we develop mathematical models and perform simulations using our MIMO Simulator with Amplifiers (MIMOSA) platform from the previous chapter.

4.1 Measurements

4.1.1 MIMO PA Testbed

To explore the effects of nonlinearities in massive MIMO as well as other topics such as cross-division duplexing (XDD) [19], we developed a state-of-the-art testbed. The platform consists of four Xilinx ZCU208 development boards [88], driving 16 PAs.. The PAs are part of the NXP Rapid RF development boards [89]. The actual PA used on the kit is an laterally-diffused metal-oxide semiconductor (LDMOS) Doherty

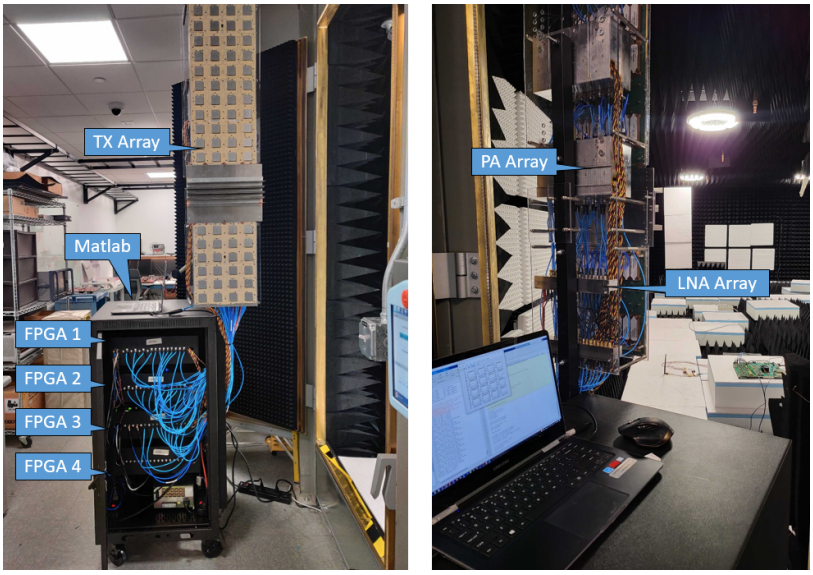


Figure 4.1: Photos of the MIMO PA Testbed.

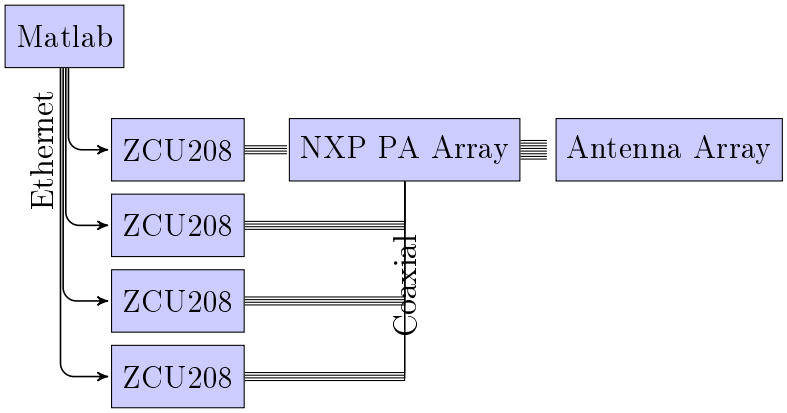


Figure 4.2: Block Diagram of Testbed. There are 12 total RF coax connections between each ZCU208 and the NXP PA array, a TX path, a RX path, and a feedback path. We support a total of 16 TX and 16 RX paths.

PA with a peak power of 43.2 dBm. These PA development boards have a dedicated feedback path that is coupled to the PA output. Such a feedback is often used for monitoring the output nonlinearities and digital predistortion (DPD) learning. A custom interface board is attached to the FMC-HPC connectors of the ZCU208 to perform the appropriate filtering and interfacing between the RF SMA connector ports and the eight digital-to-analog converters (DACs) and eight analog-to-digital converters (ADCs). We connect the PA feedbacks for each PA into RX ports on the interface boards to be sampled by the ADCs. The ZCU208 supports RF sampling on the DAC and ADC. A custom field-programmable gate array (FPGA) image was created to target the 3.5 GHz band with a maximum sampling rate of 491.52 Msps. A photo of this platform is shown in Fig. 4.1, and a block diagram is shown in Fig. 4.2.

The system can hold 10 ms of baseband I/Q data for each stream in DDR DRAM. We build an application programming interface (API) to control the testbed directly into the MIMOSA framework discussed in Section 3.1. Matlab is used to perform all signal processing, and TFTP is used to transfer the waveform into memory on each board. A trigger signal is then sent for concurrent playback and capture, where the trigger starts the data transmission from the DRAM through the sixteen DACs. It simultaneously triggers the capture from the ADCs to the DRAM. The captured data is then transferred back to MATLAB for the post-processing that follows in the remaining sections.

4.1.2 PA Variability

In this test, we seek to understand the variability that exists across an array of similar PAs. Our testbed uses the NXP PAs outlined in the previous section [90]. While the devices are identical parts, there may naturally be some variation that manifests itself in the nonlinear models. This variation would potentially mean that unique inverse

models for DPD must be created for each element in a MIMO array.

To perform this experiment, we excited each PA one at a time with a 100 MHz 5G new radio (NR) signal and sampled the output at 491.52 MHz for an approximately 5x upsample factor. Each PA was targeted at a 30 dBm output power. However, there are differences in the exact PA output powers due to unaccounted-for individual losses specific to each TX path. For example, there may be minor differences in the exact output power due to different cables, different cable lengths, different tightness of RF connectors, and natural differences in components.

In Table 4.1, we show the output adjacent channel leakage ratio (ACLR) of the 16 PAs in the testbed. We use the feedback paths in the testbed to capture the PA output signals, and we perform the measurement in Matlab. From this, it can be seen that these Doherty PAs are highly nonlinear and violate the 3rd Generation Partnership Project (3GPP) spectral emission mask of -35 dBc by 20 dB. The power spectral density (PSD) output of the PAs is shown in Fig. 4.3.

We then calculate a memory polynomial (MP) model for each PA. We consider a $P = 7, M = 1$ model and fit the input/output data using least squares. As will be shown in subsequent analysis, the phase of the MP coefficients will ultimately determine how coherently the nonlinearities combine. We show the range of the phases in the boxplot in Fig. 4.4, which shows how similar the PA models are.

From this study, we can see a strong similarity between all the models. Table 4.1 shows similar ACLR performance, Fig. 4.3 shows a similar output spectrum, and Fig. 4.4 shows similar DPD coefficients. However, to meet the stringent emission mask requirements, it is likely that unique models would need to be captured. For future work, a meaningful additional experiment would be to examine the sensitivity of the DPD performance when using an average model. If an average model can predistort effectively, then the number of ADCs for feedback DPD learning could

Table 4.1: PA Array ACLR Comparison

PA Index	L1 (dBc)	U1 (dBc)	PA Index	L1 (dBc)	U1 (dBc)
1	-25.5	-26.5	9	-26.2	-26.5
2	-24.0	-28.1	10	-26.0	-27.3
3	-25.3	-26.7	11	-25.5	-29.0
4	-24.6	-27.7	12	-24.5	-28.7
5	-25.0	-27.3	13	-27.9	-29.7
6	-26.1	-27.9	14	-25.6	-27.5
7	-24.9	-27.3	15	-26.4	-28.6
8	-26.5	-28.3	16	-27.3	-31.0

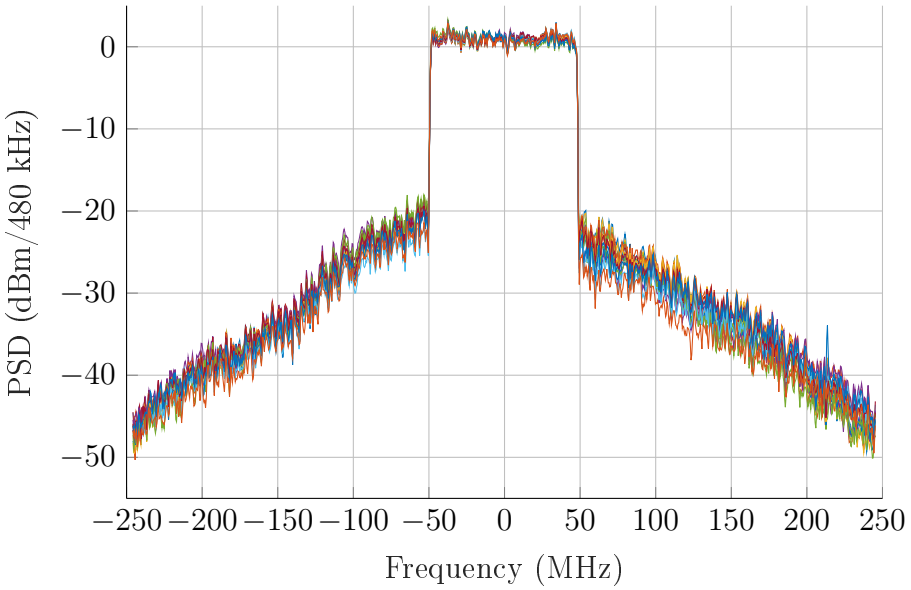


Figure 4.3: Example PSD for the set of 16 PAs. Each PA is driven with a 100 MHz 5G NR OFDM signal. From this plot, the overall relative similarity in the spectrum can be seen.

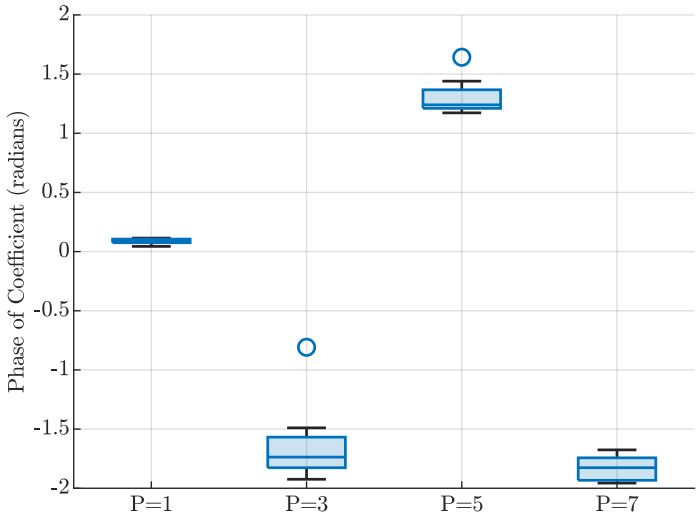


Figure 4.4: Variation of phases for each polynomial term in a $P = 7, M = 1$ memory polynomial.

potentially be reduced. Moreover, a single model could be stored in memory, reducing the implementation requirements.

4.1.3 Nonlinear Behavior over a Simulated Channel

In the previous section we observed similarities in the PA models. However, this tells us little about the behavior in an actual massive MIMO platform where it is possible to have coherent combining of signals at some points farfield of the array. In this section, we expand on the results in Section 4.1.2 to understand the behavior over-the-air (OTA).

Ideally, we would be able to capture the spectrum OTA at every point in space and report measurements on the ACLR. However, we can practically only support a finite number of observation antennas. A limited setup of a few observation receivers would result in a limited picture of the array behavior. Moreover, only a small sector of angles would be possible to observe based on the anechoic chamber geometry, where the testbed is placed. We instead opt for using the real PA array from our testbed with a simulated channel. Using a simulated channel allows us to have a complete channel

model in every direction so that we can more completely analyze the performance. With the simulated channel, we also gain the advantage of having perfect channel state information (CSI) to calculate the zero-forcing (ZF) and maximum ratio transmission (MRT) precoders, eliminating another possible source of error from the work.

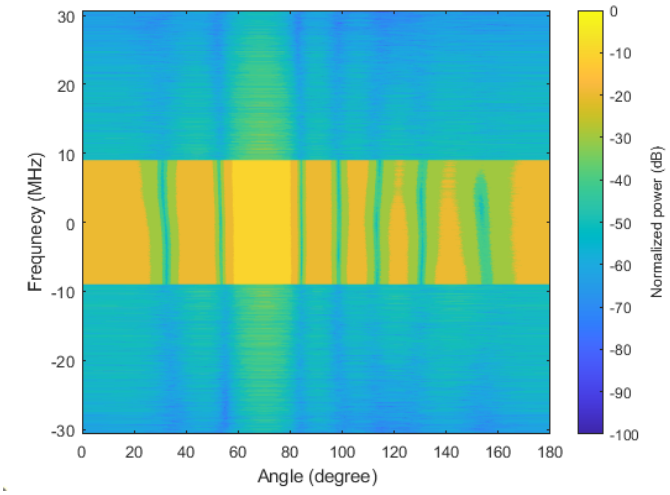
In MATLAB, we create OFDM data for one or multiple users, as outlined in Section 2.1.1. We utilize QUAsi Deterministic RadIo channel GenerAtor (QuaDRiGa) to create a line-of-sight (LoS) channel model for a uniform linear array (ULA) operating at 3.5 GHz [91]. We transmit data through the platform, similar to Section 4.1.2 and collect the PA output via the feedback ADCs. The baseband data is copied back to the host PC for post processing in Matlab.

Single User

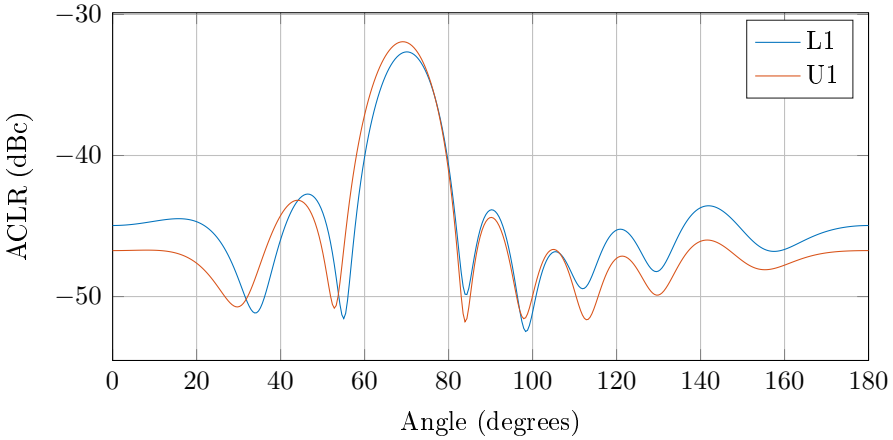
To evaluate the behavior of the adjacent channel power (ACP), we begin with beamforming to a single user. We create an LoS channel in QuaDRiGa, simulating the channel to a user placed at 70° to the axis of the array.

The results are shown in Fig. 4.5. In Fig. 4.5a, we plot the beamgrid for the data collected from the feedback of all of the PAs. This plot allows us to evaluate the angle of departure (AoD) for the inband and out-of-band (OOB) energy. The inband data is 1200, 15 kHz subcarriers. From this plot, it is clear that the OOB energy is dominant in the direction of the main beam at 70°. We then computed the ACLR for each angle. Typically, ACLR is defined as the ratio of in-band and out-of-band power. However, in this case, the in-band power is also a function of angle due to the beamforming. We therefore use the beamformed in-band power as the reference level for the calculation,

$$\text{ACLR}(\theta) = P_{\text{user}} - P_{\text{adjacent}}(\theta) \ (dB). \tag{4.1}$$



(a) Beamgrid showing the array responce in all directions.



(b) ACLR vs. angle.

Figure 4.5: Beamforming results for single-user measurement. The ACLR is highest in the direction of the user at 70° with a value of -32.7 and -31.9 dBc for the L1 and U1 adjacent channels, respectively.

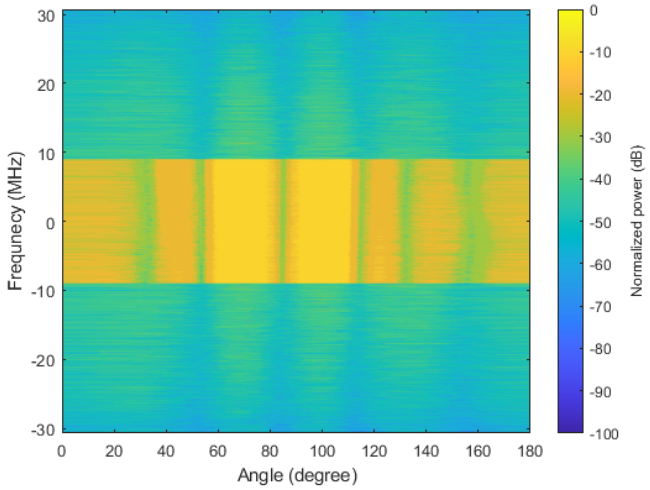
The result is plotted in Fig. 4.5b. Here, the ACLR reaches -32.7 and -31.9 dBc for the L1 and U1 adjacent channels, respectively. In other directions, the ACLR is over 10 dB lower. From this result, we can clearly see that the ACP coherently combines in the direction of the user.

MU-MIMO

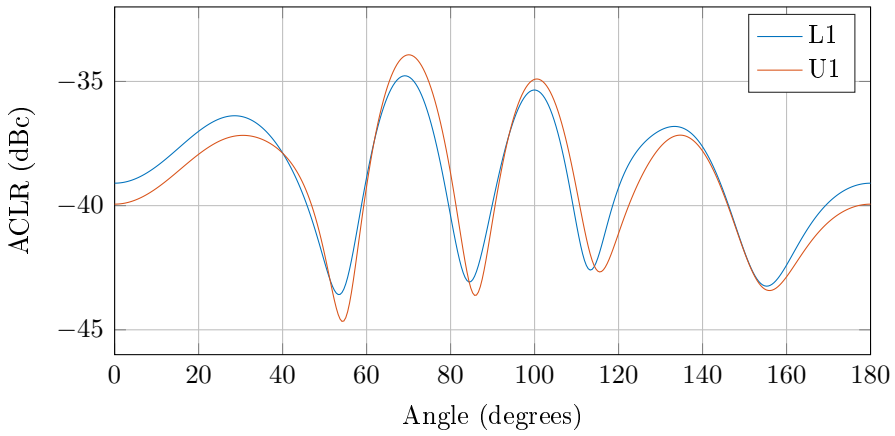
We then extend the previous experiment by beamforming to two users. We create an LoS channel in QuaDRiGa, simulating the channel to users placed at 70° and 100° to the axis of the array.

The results are shown in Fig. 4.6. In Fig. 4.6a, we plot the beamgrid for the data collected on the output of all of the PAs. The inband data is again 1200, 15 kHz subcarriers centered around 0 Hz. From this plot, it is clear that the OOB energy is dominant in the direction of the main beams at 70° and 100°. However, there are additional spurious beams that appear.

To examine the behavior more closely, we plot the ACLR versus the angle in Fig. 4.6b. Similar to the single-user case, the ACLR peaks in the direction of the users. However, additional beams are created in the multi-user (MU) case, similar to (4.9) and (4.10), which will be presented later in this chapter. The peaks are slightly different for the U1 and L1 channels due to the two channels existing at different frequencies. On average, they appear at approximately 30° and 134°. According to Eq. (4.12), these beams were predicted to appear at 30.9° and 133.6 °. The discrepancy between the predicted value and the measured value is likely due to a combination of effects. Most notably, Eq. (4.12) is based on a narrowband channel. Other possible contributors to the error include the presence of higher-order nonlinearities and memory effects. One interesting finding in this measurement is that the ACLR throughout space is better than ACLR at any particular PA, as shown in Table 4.1. This is to



(a) Beamgrid showing the array responce in all directions.



(b) ACLR vs. angle.

Figure 4.6: Beamforming results for two-user measurement. The users are at 70° and 100°. While the OOB energy is dominant in the directions of the users, a notable out-of-beam emission occurs around 30° and 134°.

be expected as the ACLR is not intentionally beamformed and hence does not experience the same beamforming gain as the inband signal. This finding suggests that lower-complexity DPD models may be sufficient for some scenarios.

4.1.4 RENEW Testing

To further examine the behavior of OOB emissions, we performed a series of measurements using RENEW. The following subsections present an overview of the platform and the testing performed.

4.1.5 RENEW Setup

RENEW is a massive MIMO platform developed for research purposes at Rice University[29]. It is constructed using a series of software-defined radios (SDRs) that are each referred to as an Iris [30]. While a real-time flow for operating RENEW exists, known as Agora, it does not easily allow for rapid prototyping as it is a large C/C++ project [92]. Instead, we opt to use the RENEWLab flow for developing a non-realtime infrastructure for our tests.

Limitations While RENEW allows for a fully programmable massive MIMO system, there are a few tradeoffs. Firstly, a limited sample rate is available. While the Lime Microsystem's radio can support up to 56 MHz of contiguous bandwidth, system reliability is affected when choosing bandwidths greater than 10 MHz. The LMS7002M also implements a digital intermediate frequency (IF). Currently, there is a strong image that is only 20 dB below the primary carrier caused by I/Q imbalance. While this is correctable through calibration procedures, this is beyond the scope of this work. The system's output power per PA is approximately 6 dBm. While the Citizens Broadband Radio Service (CBRS) radio frequency (RF) frontends that are

being used can support up to 28 dBm, the available gain is limited to protect the hardware, but can be undesirable when studying PA behavior. Finally, when using RENEWLab, we load samples into the FPGA system's BRAM. Due to limitations in BRAM, a downlink slot must be less than 4096 samples.

4.1.6 Example PA Measurement

Before understanding the MIMO behavior of the array, we first seek to characterize an example PA in the system. Using RENEWLab and MIMOSApy, we build an LTE signal with a 3 MHz channel. We then transmit out of a single Iris in the RENEW base station (BS) while connected to a calibrated spectrum analyzer. We find that the output power is 6 dBm. with an ACLR of -30.05 dBc and -30.99 dBc in the first lower and upper adjacent channels. This measurement is shown in Fig. 4.7.

4.1.7 Measurement of OOB Radiation

To perform the measurement of radiation with respect to angle, we ideally need to collect an infinite number of measurements concurrently along an arc of a constant distance from the BS. However, this is not feasible. In this section we detail our measurement methodology in the absence of a system with infinite antennas.

Firstly, a RENEWLab MIMOSApy application was written in Python to perform the following: 1. Construct a waveform for testing. 2. Construct a pilot sequence that is robust against carrier frequency offset (CFO) and other impairments. 3. Measure the channel at a remote user equipment (UE) that is not synchronized to the transmitter. 4. Precode using MRT at the BS and transmit continuously so that the spectrum can be collected.

Figure 4.7: Output Spectrum of RENEW. The RENEW PA is set to an output power of 6 dB. The ACLR is -31 dBc on each of the first adjacent channels. The true BB center frequency is seen as the impulse, and an image is present opposite of the center frequency.

Waveform Design While using RENEW, we choose to design a waveform with sufficient upsampling factor to potentially support DPD. Due to the previously stated bandwidth limitations, we used an LTE waveform designed for a 1.4 MHz channel sampled at 7.68 Msps to provide over 5x upsampling. To fit the sample constraint, we used five symbols, each with 72 subcarriers.

Time Synchronization To transmit using RENEWLab, the BS normally transmits a beacon. The UE FPGA design contains a correlator that will search for the beacon when triggered. However, we found this to be unreliable and opted for a software corrector. The BS is programmed to repeat the DL slot for a full 10 seconds. The UL is triggered to start capturing during this transmit window. Then in MIMOSApy, we perform a crosscorrelation to find the start of one DL slot.

Pilot Sequence, Channel Learning, and Precoding To learn the channel, we transmit known signals on each antenna, where each antenna transmits on a subset of distinct subcarriers. By having all antennas transmit concurrently, impairments that are time dependent such as CFO can be avoided. We postprocess the UE captured waveform to search for the pilot waveform. Once synchronized, the channel per subcarrier can be calculated, interpolating between subcarriers for a single antenna.

Spectrum Measurement Instead, we set up a mobile spectrum analyzer to record the channel power in the main and adjacent channels, integrated over the 1.4 MHz channels. We then start recording at the spectrum analyzer and walk at a constant pace in a semicircle around the array. These measurements are likely affected by the presence of the human moving through the environment holding the spectrum analyzer.

(a) Indoor Testing Scenario. (b) Outoor Testing Scenario.

Figure 4.8: RENEW OTA testing setups for measuring OOB radiation.

Indoor Beamforming We place a UE node approximately 5 m at 70 degrees from the plane of the array. The UE operates on battery power and uses a long ethernet cable as a control backhaul to the server.

Outdoor Beamforming We place a UE node approximately 20 m in front of the array. The UE operates on battery power and uses a long ethernet cable as a control backhaul to the server.

From these experiments, we can clearly see beamforming to our user of the main in-band data signal as well as the OOB emissions. These results are similar to the tests from Section 4.1.3, however are less precise due to the measurement methodology. For future study, one could perform similar tests with concurrent measurements from multiple UE nodes as well as precoding for MU-MIMO.

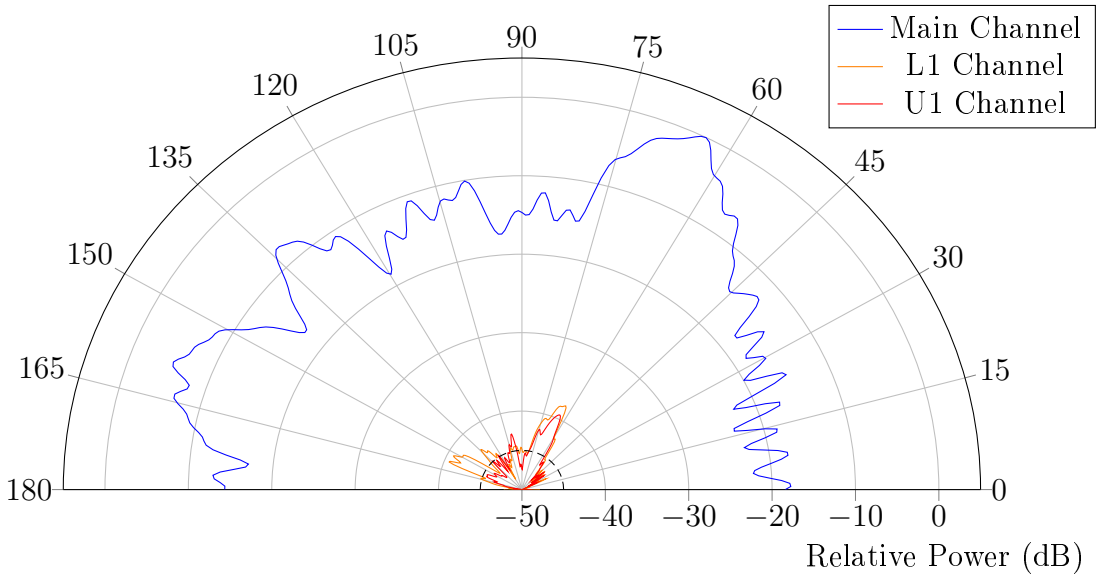


Figure 4.9: Indoor RENEW spectrum beamforming measurement where the user is approximately at 70 degrees.

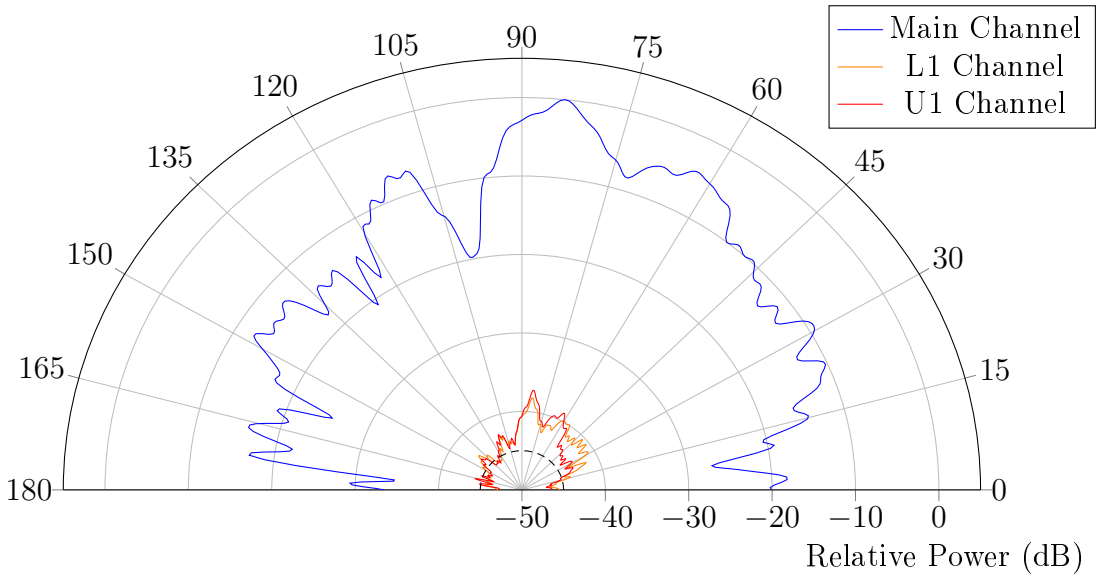


Figure 4.10: Outdoor RENEW spectrum beamforming measurement where the user is approximately at 90 degrees.

4.2 Models

4.2.1 Spatial Intermodulation

The question of how the unintended ACP is radiated in MIMO systems has seen recent attention in the literature [80], [93]–[95]. However, the analysis is often only done to some mathematical conclusions without providing any practical answer about which direction the spurious beams point. Ultimately, the answer to this question will depend on the array geometry considered and the precoding scheme. However, we will answer this question for a half-wavelength spaced uniform linear array (ULA) with MRT beamforming in an LoS channel.

To build our analysis, we consider a multi-tone signal. Consider two users with incident angles at θ_1 and θ_2. Assuming a planar wave model, the channel vector to the i-th user can be written as,

$$[\mathbf{h_i}]_n = \left[e^{-j\pi n \cos\theta_i}\right], \tag{4.2}$$

where $n = 0, ..., N-1$ indexes the base station antenna [96]. The MRT precoder, $\mathbf{p}_i$, is then formulated via the complex conjugate of Eq. (4.2).

Similar to [95], we consider the case of transmitting I/Q modulated data tone to each user. The I/Q modulation for each user is given through the amplitude and phase modulations, $A_i(k)$ and $\gamma_i(k)$. After precoding, the composite baseband signal at each PA input would be,

$$x_n(k) = A_1(k)e^{j(\gamma_1(k)+\phi_{1,n})} + A_2(k)e^{j(\gamma_2(k)+\phi_{2,n})}, \tag{4.3}$$

where $\phi_{i,n}$ is the phase shift due to precoding for the i-th user on the n-th antenna,

given as

$$\phi_{i,n} = \pi n \cos\theta_i. \tag{4.4}$$

One popular model used for behavioral modeling and predistortion is the MP and its extension, the generalized memory polynomial (GMP)[42]. The MP is given as,

$$\hat{x}(k) = \sum_{p=1}^{P} \sum_{m=0}^{M} \alpha_{p,m} x(k-m) \left|x(k-m)\right|^{p-1}, \tag{4.5}$$

where P is the maximum nonlinear order, M is the number of memory taps, k is the sample index, and $\alpha_{p,m}$ is a complex scalar coefficient corresponding to a particular device.

While in Section 4.1.2, high-order MP PAs with memory are used, in our analysis, we consider the following memoryless third-order model for mathematical tractability,

$$\hat{x}(k) = x(k) + \alpha_n x(k)|x(k)|^2, \tag{4.6}$$

where α_n is the complex coefficient specific to the model of PA n. After Eq. (4.3) is substituted into Eq. (4.6), we get the output of each PA as

$$y_n(k) = \left(A_1 + \frac{3\alpha_n}{2} A_1 A_2^2 + \frac{3\alpha_n}{4} A_1^3\right) e^{j(\gamma_1 + \phi_{1,n})} \tag{4.7}$$

$$+ \left(A_2 + \frac{3\alpha_n}{2} A_1^2 A_2 + \frac{3\alpha_n}{4} A_2^3\right) e^{j(\gamma_2 + \phi_{2,n})} \tag{4.8}$$

$$+ \frac{3\alpha_n}{4} A_1 A_2^2 e^{j(2\gamma_2 - \gamma_1 + 2\phi_{1,n} - \phi_{2,n})} \tag{4.9}$$

$$+ \frac{3\alpha_n}{4} A_1^2 A_2 e^{j(2\gamma_1 - \gamma_2 + 2\phi_{2,n} - \phi_{1,n})}. \tag{4.10}$$

In the above, the sample index, k, is dropped from the amplitude and phase modulation terms for compactness. The goal is then to find the physical directions with

respect to the array, $\hat{\theta}$, that correspond with the spurious beams caused by the terms from (4.9) and (4.10). Without loss of generality, we focus on the term from (4.9) and look for the angle that maximizes the array response,

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \sum_{n=0}^{N-1} \frac{3\alpha_n A_1 A_2^2}{4} e^{j(2\gamma_2 - \gamma_1 + 2\phi_{1,n} - \phi_{2,n} - \pi n \cos\theta)}. \tag{4.11}$$

In cases where all PAs have identical phases on α_n, the coefficient can be eliminated as the phase shift would be common across all elements in the array. Otherwise, as they become more randomly distributed, the less coherently the intermodulations combine [95]. To solve the above, we note that the beamforming direction does not depend on the sample index, k, and set it to zero. For each $A_i(k = 0)$, we have a real-valued scalar. Being without phase, it has no effect on the beamforming direction. For the phase modulation terms, $\gamma_i(k = 0)$, we have a constant phase shift applied uniformly across all elements, so it too can be disregarded. We note that the sum will be maximized when, if possible, all the exponential terms are cophased. We choose to force the phase of each exponential term to be zero and solve for the θ that allows for that. With the above assumptions and replacing the ϕ terms with their exact values from Eq. (4.4), we arrive at the direction of interest for the term from (4.9) as,

$$\hat{\theta} = \cos^{-1}\left(2\cos\theta_1 - \cos\theta_2\right). \tag{4.12}$$

A similar result was first published in [97]. However [97] and other works do not consider the case where the inverse cosine is undefined. While [97] states that any intermodulation falling outside the window of the 0–180 degrees will not radiate, we find through our MIMOSA numerical simulations that these beams in fact wrap back around. To ensure that the previous equation falls within the domain of the inverse

cosine, the following can be performed.

$$\hat{\theta} = \cos^{-1}\left(2\cos\theta_1 - \cos\theta_2(\mathrm{mod}2) - 1\right). \tag{4.13}$$

The above has been empirically verified against our simulation platform.

While the two-tone analysis is simple, the conclusions scale up similarly to OFDM [95]. When scaling up to more than two users, the spurious beams will scale up with every pair and triple of users. The main idea is to find the number of terms similar to (4.9) and (4.10) as more users are added. From analysis similar to deriving (4.9) and (4.10), it can be shown that there will be a third-order intermodulation (IM3) term for each pair and triple of user signals. In the case of the previous two-tone, a three-user system will have terms with phases of the form as $\phi_{1,n} + \phi_{2,n} - \phi_{3,n}$. This triple can be arranged in three unique ways, and, similarly, each pair of terms can be arranged in two ways. Combining these ideas, the upper bound on the total number of spurious beams created by the IM3 between user streams is given as,

$$n_{\text{spurious beams}} = 2\binom{U}{2} + 3\binom{U}{3}. \tag{4.14}$$

This expression is considered an upper bound since, for specific user angles, the spurious beams may overlap. For example, when users are regularly spaced at some angle, these spurious beams may be in the same direction and appear as a single beam. There will be more terms and spurious beams for higher-order nonlinearities, but the third-order intermodulations are typically the highest magnitude and, hence, are the primary concern.

4.2.2 Complexity of DPD per Antenna

A primary motivation for this work is to reduce the complexity of the predistortion for fully-digital massive MIMO arrays. In this section, we develop the baseline for the application complexity of a GMP-based DPD [42] per antenna.

There are various design tradeoffs that could be deployed in practice when implementing a DPD. For 5G systems with wide bandwidths, the overall throughput will be most critical. So in this analysis, we consider a fully pipelined design with maximum parallelism for a single stream. In practice, if a given clock rate can not support the desired sample rate, the designer may further parallelize to where multiple output samples are computed in parallel per clock cycle. However, this dramatically increases the area requirements of the design.

From Eq. (2.3), it can be seen that there are three possible loops, the polynomial order, the memory taps, and the lag/lead. The main overall structure of the design is shown in Fig. 4.11. Each polynomial "branch" of the memory polynomial corresponding to nonlinear order p computes $x(n)|x(n)|^{p-1}$, and there is a branch for each p in the design. This computation from each branch is passed to an finite-impulse-responce (FIR) filter with complex taps. Three multiplications are used for each complex multiplication in each filter. We will use these assumptions when computing complexity comparisons in Chapter 5.

4.3 Simulations

While the mathematical models developed in the prior section provide valuable insight, it becomes intractable to answer many important questions. For MU-MIMO with OFDM and high-order polynomials, deriving closed-form expressions would not be beneficial as the necessary complexity in the model would obfuscate any possible

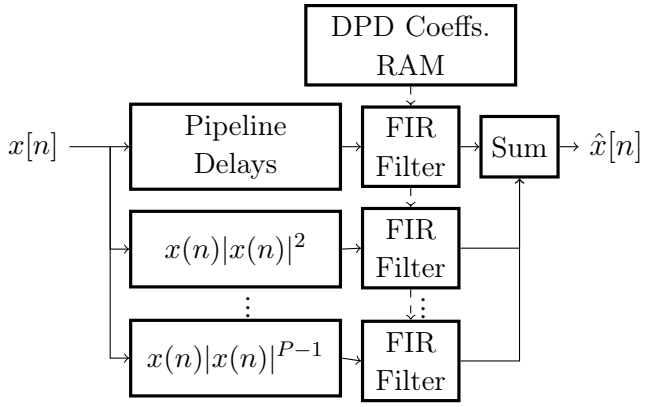


Figure 4.11: General structure of the high-throughput, low-latency, memory polynomial FPGA implementation. Adapted from [12].

conclusions.

To solve this problem, we utilize MIMOSA and MIMOSA for Python (MIMOSApy) from Chapter 3. Here, users can be arbitrarily placed in an environment, and a channel model can be generated via QuaDRiGa, using the 3GPP 38.901 rural macrocell (RMa) model, for example. A ULA can be created with arbitrary size and with arbitrary PA models at each antenna. Then OFDM processing can be performed for each user with MRT or ZF precoding. This simulation platform can provide tremendous insight on the expected behavior of a realistic MIMO system.

For these simulations, we include a memory polynomial measured in Section 4.1.2 as the default polynomial coefficients. In cases where we introduce variability, we add the specified variability to each coefficient to create a random Gaussian distribution in the coefficient centered around the measured value.

4.3.1 Simulation Example

With the simulation platform, a wide variety of data and plots can be generated for each test. However, for each of the subsequent tests, we can not show the full set of information as this would be overwhelming. In this subsection, we provide a

Table 4.2: Nonlinear MIMO Link-level Simulation Parameters

Parameter	Value
Number of Antennas	64
Number of Users	4
User Locations (degrees)	70, 85, 100, 125;
PA Order, PA Memory	7, 4
PA Variability	10%
Channel	QuaDRiGa 3GPP 38.901 RMa
Precoding	ZF
Center Frequency	3.5 GHz
Active Subcarriers	3240
Subcarrier Spacing	30 kHz

Table 4.3: Nonlinear MIMO Link-level Simulation Results

Parameter	Value
Average PA ACLR	-26.5
Worst Far-field ACLR	-30.9
Number of violating directions	19

complete set of results for one representative scenario. We then dissect each aspect into simplified tests and present the relevant metrics that provide the most insight.

In Table 4.2, we show the simulation parameters for this test. In Fig. 4.12, we show the beamgrid plot. Here, the primary beams and nonlinearity can be seen at 70°, 85°, 100°, and 125°. Throughout the angular domain, various other peaks appear.

4.3.2 ACLR Versus the PA Variability

In this section, we focus on the effect of PA variability. We consider a MP PA where the phase of the third-order coefficients are allowed to vary by various amounts. For this particular model, the ACLR of each PA is fixed to approximately -33 dBc in all tests. While the ACLR of each PA remains constant, we see in Fig. 4.14 that the effective ACLR in the farfield decreases. As the phase varies, the third-order term

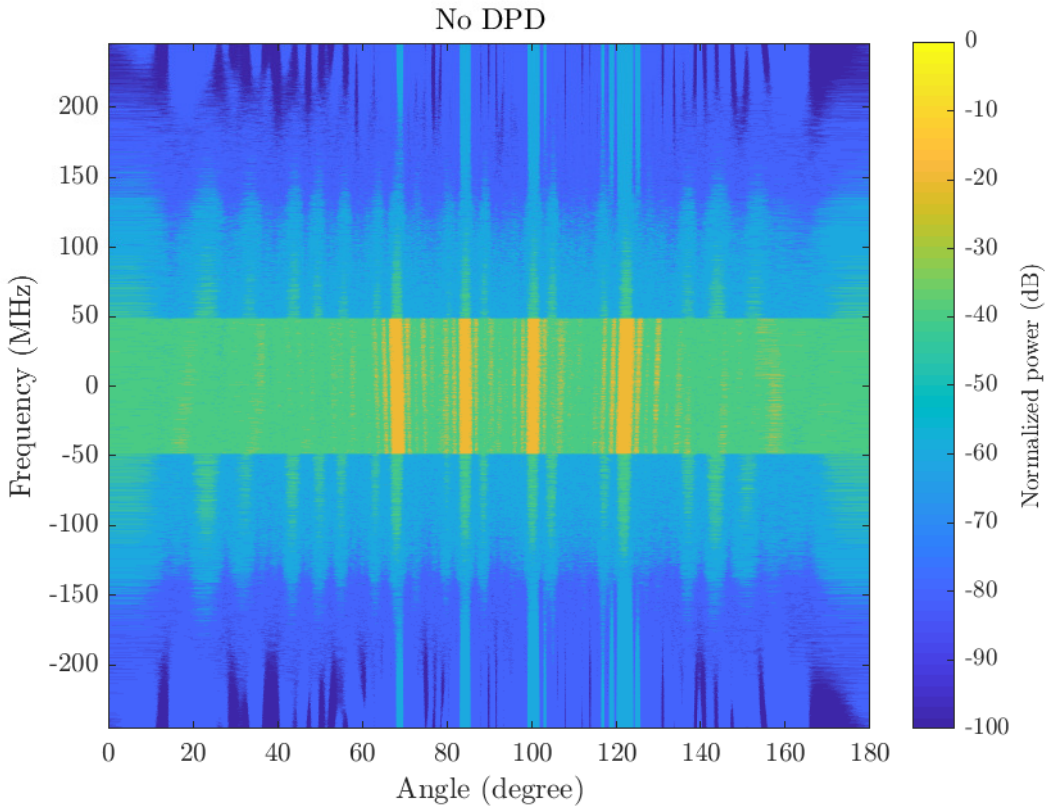


Figure 4.12: Example beamgrid showing the energy in each resource element and AoD from the array for the four user parameter case shown in Table 4.2.

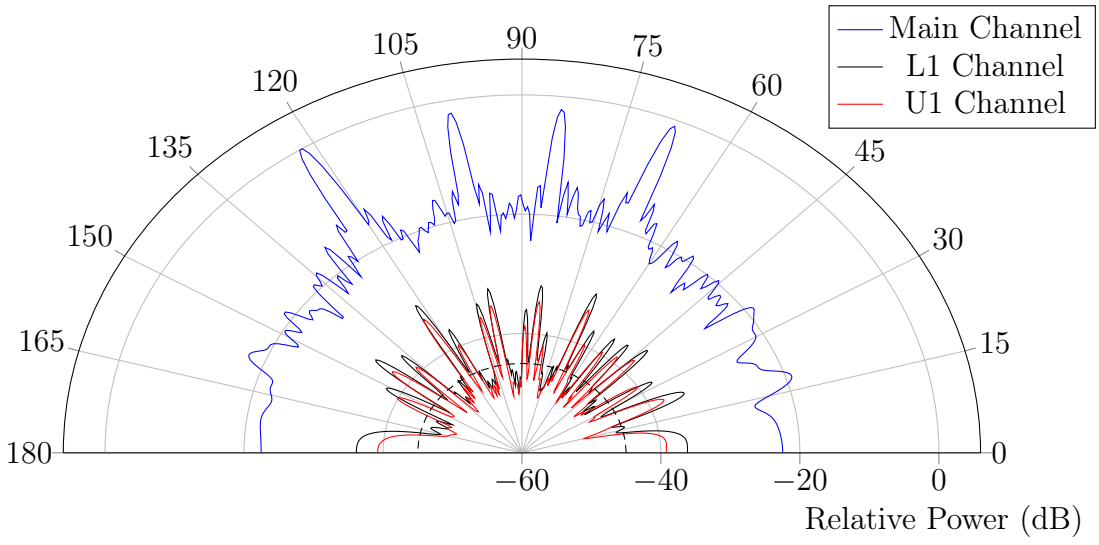


Figure 4.13: Example beamplot of integrated channel powers for the four users at 70°, 85°, 100°, and 125°.

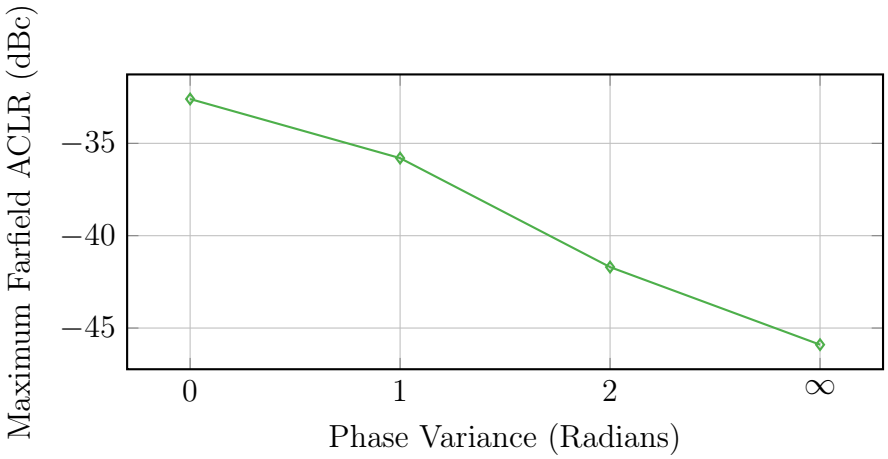


Figure 4.14: As the variance in the phase of the coefficients increases, the maximum farfield ACLR decreases.

sees less of the array gain, confirming our conjecture.

4.4 Conclusion

In this chapter, we performed a variety of initial explorations to better understand the effect of nonlinearities in a massive MIMO array. Firstly, we measured the variability in an actual Doherty PA array and found that the coefficients were overall similar. When projecting this data through a simulated channel, we found that the OOB emissions would coherently combine. We then showed that a third-order nonlinearity would coherently combine not only in the direction of users, but also in other directions that were mathematically predictable. We then verified this finding using MIMOSA for a MU-MIMO scenario with four users. In the following chapter, we will explore a novel DPD scheme for MU-MIMO that is performed before the precoder.

Chapter 5

Virtual DPD Solutions

In massive multiple-input, multiple-output (MIMO) arrays, there may be hundreds of active elements. Using traditional digital predistortion (DPD) schemes, we would linearize each, which creates a challenging computational burden. Based on the analysis done in the previous chapters, we seek in this chapter to explore new predistortion methods tailored to massive MIMO. In particular, we seek to move the predistorter to exist before the precoder, creating a novel scheme that we call virtual DPD (vDPD). By placing the predistorter before the precoder, we relax the DPD requirement per base station (BS), allowing it to scale, in principle, with the number of users instead of the number of antennas. To undertake this challenging task, we embarked on a three-step process. Firstly, in Section 5.1, we worked on developing a DPD system that could work in the frequency domain for single-input, single-output (SISO) systems. This compatibility with frequency domain processing is necessary because most precoders in modern cellular systems are frequency-domain based for orthogonal frequency-division multiplexing (OFDM). Secondly, in Section 5.2, we scale this idea to operate in a multiple-input, single-output (MISO) case, where many antennas transmit to a single-user. Finally, in Section 5.3, we look at the case of a massive multi-user (MU)-MIMO array where we improve the out-of-band (OOB) emissions

not only in the user beams but also for any spurious beams.

5.1 vDPD for Single Antenna — ODPD

[1] In 5G and other popular radio access technologies (RATs), the baseband signal is constructed in the frequency domain as OFDM. Most DPD schemes operate in the time domain, completely agnostic to the modulation scheme. While this allows most DPDs algorithms to be highly portable across systems, it is problematic for easy integration before frequency-domain precoders used in massive MIMO. In this section, we address this problem in the single-antenna case by developing a scheme that operates on individual subcarriers, with the goal of portability to before the precoder in MIMO in later sections of this chapter.

This section introduces OFDM DPD (ODPD), a novel DPD method for OFDM waveforms that exploits the guard-band subcarriers typically present in OFDM-based systems. In particular, instead of transmitting zeros on the guard-band subcarriers, we iteratively tune their values to reduce the adjacent channel leakage ratio (ACLR) on a per-OFDM-symbol basis. To determine the appropriate values of the guard-band subcarriers, one needs only a forward model of the power amplifier (PA), as opposed to the inverse model needed in most DPD solutions. Our experimental measurements using a commercial Doherty PA platform shows that we can achieve linearization that outperforms a state-of-the-art polynomial model. Moreover, when combined with an neural network (NN)-based PA model, our OFDM-based DPD can perform DPD with as little as a 2x upsampling/oversampling rate[2] as opposed to the 5x upsample rate typically considered in polynomial-based DPD solutions [42] [5]. The lower sampling

[1] This work was originally presented in [14].

[2] While upsampling refers to the digital signal processing (DSP) process of interpolating and oversampling refers to the relative rate of the analog-to-digital converter (ADC) sampling rate to a signal's Nyquist rate, we will use the terms interchangeably throughout this section for brevity.

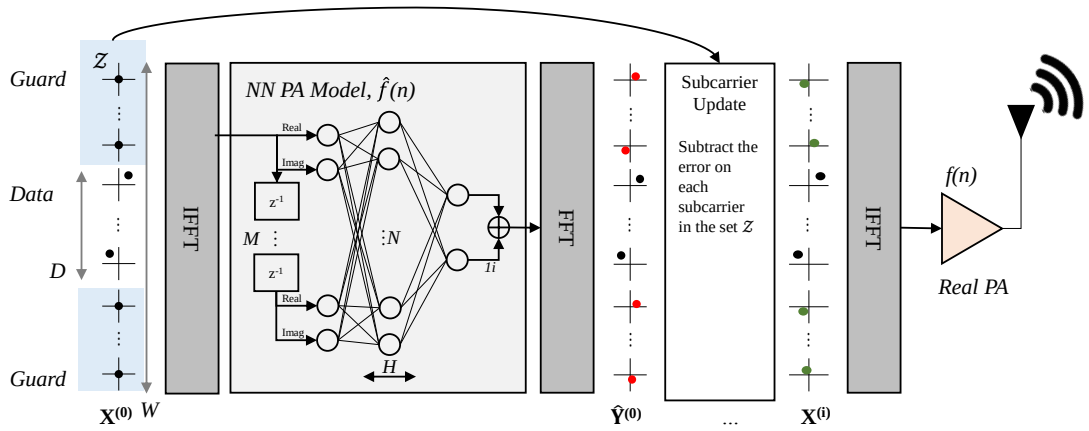


Figure 5.1: Overview of ODPD algorithm. For each symbol, we transmit through a PA model, $\hat{f}$, to see the error in the guard-band subcarriers in the set $\mathcal{Z}$. The error is subtracted to form the new frequency-domain input. After sufficient iterations, the symbol can be transmitted through the real PA. In this figure, we omit the DAC and up-converter after the final IFFT for simplicity.

rate translates into lower energy consumption and reduced system complexity.

Similar ideas can be found in the literature, though not directly applied to DPD. For example, a similar technique is used for peak-to-average power ratio (PAPR) reduction in [98]. In [99], the authors also utilize the guard bands for *cancellation carriers*, but their goal is OFDM sidelobe suppression and not correction of PA non-linearities. The learning iterations of our method are similar to the iterative learning control (ILC) DPD method used in [58]. However, in our method, we adapt it to operate directly on the guard-band subcarriers in the frequency domain.

5.1.1 ODPD Algorithm

Let $\mathbf{X} \in \mathbb{C}^W$ denote a vector of symbols that correspond to one OFDM symbol. $W \in \mathbb{N}$ denotes the total number of subcarriers, which is typically a power of two for efficient fast Fourier transform (FFT) computations. In OFDM systems, $D \in \mathbb{N}$ where $D < W$ subcarriers carry information, and there are typically $K = W - D$ subcarriers that map to the edge of the spectrum, which are zero-filled and which form

the *guard band*. Let the set of zero-filled subcarriers in the guard band be denoted by $\mathcal{Z} \subset \{0, \ldots, W-1\}$. Our proposed method's key idea is to replace the zero-filled subcarriers with tuned values that depend on the remaining D data subcarriers to reduce the OOB emissions directly.

For simplicity, we restrict our description to a single OFDM symbol. However, our method can be extended to multiple symbols by applying ODPD for each symbol and relying on the windowing technique typically applied in OFDM systems to improve the spectrum at symbol boundaries [25]. Let $f(\cdot)$ denote the baseband equivalent of the nonlinear PA transfer function. Then, the frequency-domain output of the PA, denoted by $\mathbf{Y}$, is [1]

$$\mathbf{Y} = \mathrm{FFT}\left(f\left(\mathrm{IFFT}(\mathbf{X})\right)\right). \tag{5.1}$$

Assuming that an estimate of the PA transfer function, $\hat{f}(\cdot)$, is created, we can iteratively estimate for each subcarrier $k \in \mathcal{Z}$ the PA output, $\hat{Y}_k$, and predistort it to cancel out the tone. By iterating on each subcarrier, we heuristically can think of this as injecting the subcarrier with the energy of the opposite phase so that there can be a net cancellation. However, the exact value depends on the intermodulations of all subcarriers. Hence multiple iterations may be needed to account for new intermodulations due to previous iterations. At iteration $i \in \{0, \ldots, I-1\}$, in our proposed method, we first use (5.1) to calculate $\hat{\mathbf{Y}}^{(i)}$ based on $\mathbf{X}^{(i)}$. Then, we calculate the differences between each $\hat{Y}_k^{(i)},\ k \in \mathcal{Z}$, and the desired output $\hat{X}_k,\ k \in \mathcal{Z}$. Finally, we adapt each guard tone $X_k^{(i+1)},\ k \in \mathcal{Z}$, as follows

$$X_k^{(i+1)} = X_k^{(i)} - \mu \hat{Y}_k^{(i)}, \quad \forall k \in \mathcal{Z}, \tag{5.2}$$

[1] The cyclic prefix (CP) is also added after the IFFT and removed before the FFT. We do not model the CP here, though a 4.7 μs CP is used in the final results.

where μ is a step size and $\mathbf{X}^{(0)} = \mathbf{X}$.

The PA model $\hat{f}(\cdot)$ can be constructed through various methods. For example, a generalized memory polynomial (GMP) or a NN [12] may be attractive solutions. Moreover, in certain time-division duplexing (TDD) systems, it could be possible to train symbols using the actual PA while the radio is listening. While in this work we highlight the use of an NN-based PA model, we briefly discuss a GMP implementation in the following subsection.

GMP PA Model

The GMP from (2.3) can be used as a forward model of the PA, $\hat{f}(n)$. When using the GMP, a least-squares model can also be used to learn the set of parameters. However, contrary to the indirect learning architecture (ILA) approach, the forward model is not as susceptible to noise. A fundamental limit of ODPD performance is the accuracy of, $\hat{f}$. Hence, sufficient upsampling would be required to avoid aliasing of high order terms when using a GMP.

Neural Network PA Model

While high-order polynomials suffer from aliasing when using low sample rates [5], [42], NNs do not necessarily have the same limitations as they are model-free.

We consider a multilayer feedforward NN with H hidden layers and N neurons in each hidden layer. M time-domain inputs are given to the network to account for memory effects in the PA. For each sample, the real and imaginary components enter the NN on separate neurons. The general architecture is shown within the ODPD system in Fig. 5.1.

Let g denote a nonlinear activation function, and let $\mathbf{W}_i$ and $\mathbf{b}_i$ denote the weights matrices and bias vectors corresponding to the ith layer in the NN. The output of

the first hidden layer at time instant n is

$$\mathbf{h}_1(n) = g\left(\mathbf{W}_1\begin{bmatrix}\Re(x(n))\\ \Im(x(n))\\ \vdots \\ \Re(x(n-M+1))\\ \Im(x(n-M+1))\end{bmatrix} + \mathbf{b}_1\right). \tag{5.3}$$

The output of hidden layer $i \geq 2$ is

$$\mathbf{h}_i(n) = g\left(\mathbf{W}_i\mathbf{h}_{i-1}(n) + \mathbf{b}_i\right). \tag{5.4}$$

Finally, the output of the network after hidden layer H is

$$\hat{\mathbf{x}}(n) = \mathbf{W}_{H+1}\mathbf{h}_H + \mathbf{b}_{H+1}, \tag{5.5}$$

where the first and second elements of $\hat{\mathbf{x}}$ represent the real and imaginary part of the signal, respectively. The NNs can be efficiently realized as a series of matrix-vector multiplies. Complexity remains low when considering a rectified linear unit (ReLU) activation function, shown in Eq. (5.6), which can be implemented with a simple multiplexer. To further reduce the computational burden, a designer could consider options such as pruning and quantization,

$$\mathrm{ReLU}(x) = \max(0, x). \tag{5.6}$$

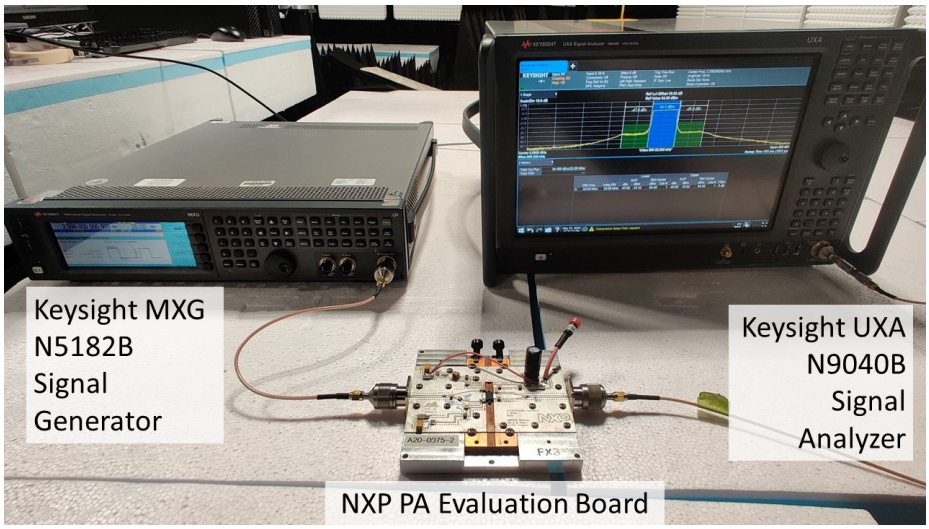


Figure 5.2: Photo of the measurement setup. A signal generated in MATLAB is uploaded to the signal generator, where it is transmitted at 3.5 GHz through the PA evaluation board into the UXA signal analyzer where the ACLR is measured.

5.1.2 Computational Complexity

The computational complexity of the ODPD scheme can be divided into two components, the complexity of the iterative application of IFFTs and FFTs, and the complexity of the forward model, $\hat{f}$. We consider the number of real multiplications as a proxy for the complexity of the ODPD and count them as

$$n_{\text{mults, FFT}} = 4IW\log_2(W), \tag{5.7}$$

$$n_{\text{mults, ODPD}} = n_{\text{mults, FFT}} + n_{\text{mults},\hat{f}}. \tag{5.8}$$

Here, $n_{\text{mults, FFT}}$ counts the number of multiplications due to the added FFTs and $n_{\text{mults},\hat{f}}$ is the forward model complexity. We assume four real multiplies per complex multiply.

Table 5.1: ACLR Measurements after ODPD

Case	L1 (dBc)	Main (dBm)	R1 (dBc)
No DPD	-30.8	35.3	-30.6
GMP			
$U = 2$	-30.5	35.2	-29.6
$U = 3$	-37.8	35.2	-36.4
$U = 4$	-41.3	35.2	-41.0
$U = 5$	-41.3	35.2	-41.3
NN, $U = 2$			
$N = 10, I = 1$	-37.2	35.2	-36.7
$N = 10, I = 2$	-38.0	35.2	-37.8
$N = 20, I = 1$	-39.1	35.3	-38.7
$N = 20, I = 2$	-40.1	35.2	-40.3
$N = 40, I = 1$	-41.7	35.2	-41.2
$N = 40, I = 2$	-42.7	35.1	-43.1

5.1.3 Results

In this section, we present experimental results to showcase the performance of our proposed ODPD method, and we compare it with a standard polynomial-based DPD method [1].

An example measurement using the ODPD method with a $N = 40$ NN is shown in Fig. 5.3 where $I = 1$. The input power for this test was 6 dBm, the in-band PA output power was 35.3 dBm (which corresponds to 29.3 dB gain from the PA). Table 5.1 shows the full measurement results for each DPD. Here, we show that the Doherty PA was highly nonlinear with a starting ACLR of -30.8 dBc on the left adjacent carrier. At a low oversampling rate of $U = 2$, the GMP is unable to resolve the high-order nonlinearities leading to an overall poor fit that causes the ACLR to

[1] Achieved ACLR performance is limited in this experiment due to measuring over the full 20 MHz without guard bands (as opposed to measuring just in 18 MHz) and lack of windowing between symbols.

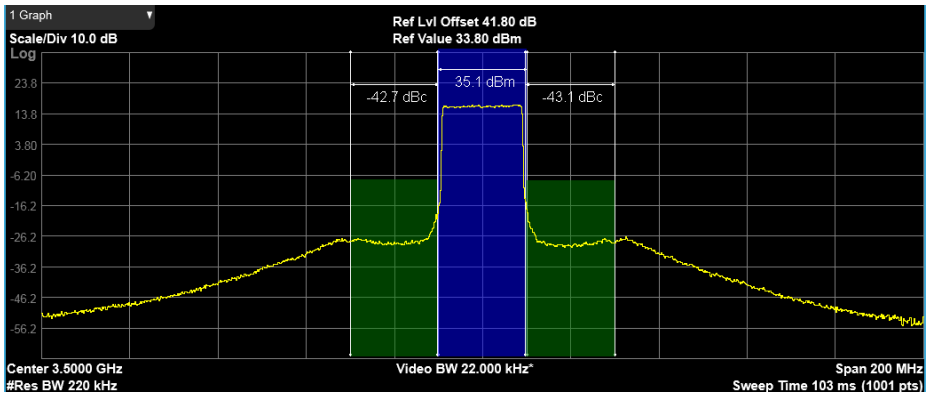


Figure 5.3: Measurement result from the Doherty PA for the ODPD with $I = 2$ iterations and $N = 40$ neurons. The blue is the 20 MHz associated with the main carrier, while the green on the left and right correspond to the 20 MHz adjacent channels.

degrade. At $U \geq 3$ the ACLR was able to improve with the GMP.

However, the NN-based ODPD was able to improve the ACLR for each considered architecture while only using an oversampling rate of $U = 2$. Moreover, the $N = 40$ architecture is able to outperform the best $U = 5$ case from the GMP.

The performance of the ODPD depends on the precision of the forward model. However, there is a tradeoff between complexity and precision. Therefore, careful tuning of the forward model is necessary. While there are many viable architectures of NNs that can be tested, we restrict our analysis to three. We restrict the NN so that $M = 4$ and $H = 1$ and vary the number of neurons in the hidden layer to be 10, 20, 40. Using the PA input/output data collected from the testbed, we train each NN in MATLAB.

In Fig. 5.4, we show the complexity in terms of the number of real multiplications per OFDM symbol of the ODPD algorithm for the three considered NNs with $I = 1$ and $I = 2$ total iterations. We compare this to the GMP ILA-DPD application complexity in red which is obtained by summing the number of multiplications in (2.3). The ODPD cases include the complexity of each new FFT and the PA model. The GMP ILA-DPD case includes the complexity of the GMP as well as the upsampling

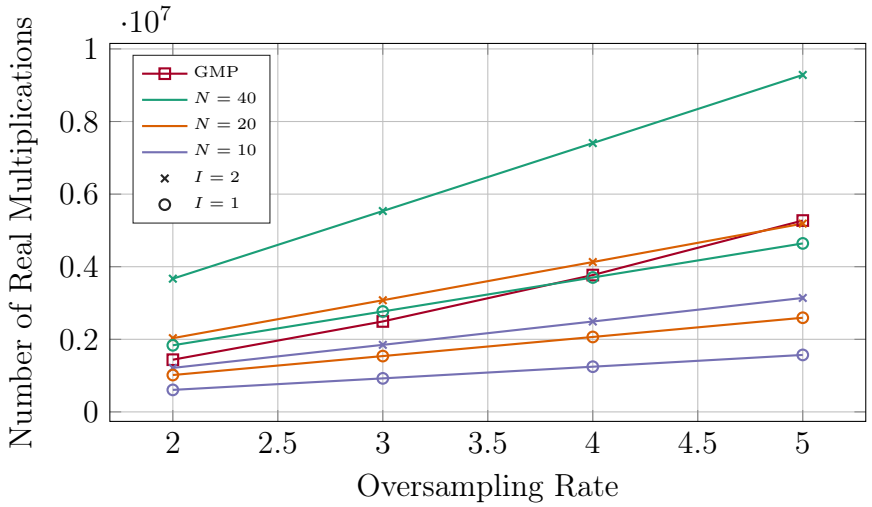


Figure 5.4: Complexity per symbol versus oversampling rate. Each NN uses $M = 4$ and $H = 1$. The GMP uses $P = 7, M = 4, L = 1$.

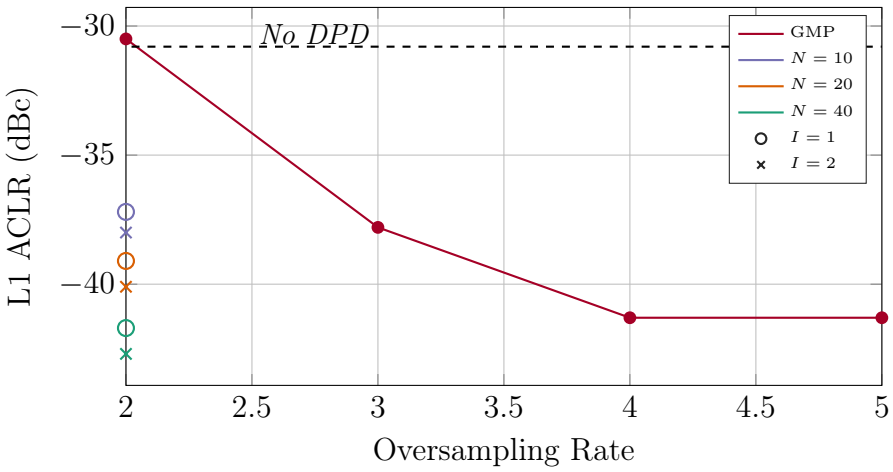


Figure 5.5: ACLR performance for each scheme. The NN-based ODPD is able to achieve an ACLR improvement with only 2x upsampling.

complexity. Upsampling is assumed to be done via filling with zeros and passing through a low-pass-filter with 51 taps.

Fig. 5.5 shows the performance from Table 5.1 for the sake of comparison to Fig. 5.4. When trying to get the most performance per computation, there are a few takeaways. Firstly, the performance of the $U = 5$ GMP can be matched by a $U = 2$ NN-based ODPD with 34.8% the number of multiplications. Secondly, it can be seen in Fig. 5.4 that for the considered NN architectures, a larger NN architecture was often less computationally intensive than more ODPD iterations on a less complex NN. It can be seen in Fig. 5.5 that the more complex NN with $I = 1$ gave a better result than the less complex NN with $I = 2$. Hence, improving the NN (or, more generally, the forward PA model) is more worthwhile than performing additional ODPD iterations.

5.1.4 Summary of Single Antenna ODPD

In this section, we introduced an OFDM-based DPD (ODPD) method that takes advantage of the guard-band subcarriers to predistort in the frequency domain. Our proposed method of predistortion does not require the estimation of an inverse PA model and was able to linearize our test PA as effectively as state-of-the-art methods. Using an NN-based forward model, we showed that this performance could be achieved with 34.8% fewer multiplications and a lower oversampling rate for the DPD application.

5.2 vDPD for Single User/Many Antenna

[1] The next step along our quest for a massive MIMO DPD scheme was a massive MISO DPD scheme. In this section, we translate the work from the previous section to exist in a many antenna system where the predistortion is performed before the

[1] A version of this work was originally presented in [16].

precoder. In the mathematical and simulation investigations, we showed that the nonlinearity follows the main beam in the case of a single beam. In this section, we treat that effective nonlinearity along that beam as if it were a single virtual PA (vPA) and linearize that vPA with a vDPD scheme based on the previous OFDM-DPD algorithm.

5.2.1 System Model and Algorithm

We consider a single-user massive MIMO system with one receive antenna at the user and N transmit antennas at the BS. Without loss of generality, we restrict the presentation below to one OFDM symbol. The data to the user is represented by the signal vector $\mathbf{s} \in \mathcal{O}^W$, where W indicates the total number of tones in the OFDM symbol and $\mathcal{O}$ represents the set of complex-valued constellation points. Pulse shaping is applied via the inclusion of guard-band subcarriers that are normally empty. We denote the set of guard subcarriers as $\mathcal{Z}$ and set $s_w = 0 \; \forall w \in \mathcal{Z}$.

Precoding is applied separately to each OFDM tone, generating W vectors $\mathbf{x}_w \in \mathbb{C}^N$. Each vector is remapped to contain all the tones per antenna, $[\mathbf{x}_1, ..., \mathbf{x}_W] = [\mathbf{a}_1, ..., \mathbf{a}_N]^T$, where each $\mathbf{a}_n$ is a W-dimensional vector containing all tones for antenna port $n \in \{1, \ldots, N\}$. At this point, the data is converted from the frequency domain to the time domain via the inverse discrete Fourier transform (DFT), which is typically calculated via an IFFT. The data is reorganized to be serial instead of parallel, and a cyclic prefix is added. In many systems, windowing is also applied between symbol boundaries to improve the spectral shaping [100]. We express this time-domain representation for each antenna as the vector $\mathbf{u}_n$. This vector is up-converted to an RF frequency where it is transmitted through a PA with nonlinear function $f_n(\cdot)$. The time-domain data for each antenna is given as $\hat{\mathbf{u}}_n = f_n(\mathbf{u}_n)$. The frequency-domain equivalent is given as $\hat{\mathbf{x}}_n$.

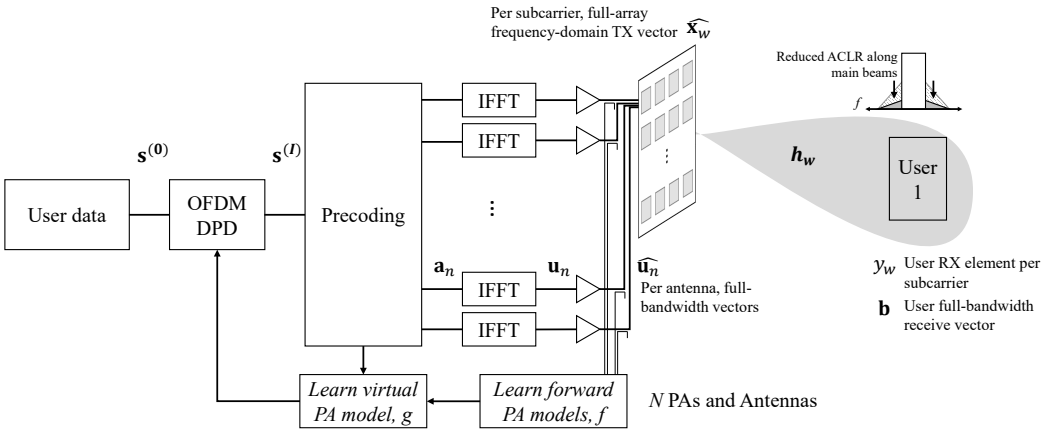


Figure 5.6: Block diagram for the beamformed, single-user vDPD system. User data is updated before the precoder with the goal of sending ACP cancellation tones in the guard-band subcarriers.

In OFDM systems, the channel is usually modeled in the frequency-domain for each tone w as, $y_w = \mathbf{h}_w \hat{\mathbf{x}}_w + n_w$, where y_w denotes the received data for OFDM tone w and $\mathbf{h}_w$ is the $1 \times N$ channel vector, and n_w is a Gaussian random noise term. The user received signal can be remapped to $[y_1, ..., y_W] = [\mathbf{b}]$ to represent a W dimensional vector of all tones received at the user. The time-domain user-received signal is given as $\mathbf{v}$.

The system architecture of our proposed OFDM-based massive MIMO DPD approach is illustrated in Fig. 5.6. Our method's main idea is to utilize the normally empty subcarriers to reduce the adjacent channel power (ACP) by injecting tones with the opposite phase of the ACP.

Virtual Power Amplifier

While in the SISO case from Section 5.1, there was a single PA that could be used in the ODPD update, in many-antenna systems we have many PAs transmitting to a user. To adapt the ODPD idea, we seek to create a composite model of the nonlinearity experienced by the user, which we refer to as a *virtual PA* (vPA). To do this, the transfer function of each PA is modeled with a memory polynomial from

Eq. (2.3). While this is similar to the memory polynomial DPD approach, these are forward models which are shown to be less susceptible to noise [61] and will only be used in this learning phase of our algorithm.

Given data for a user, we can perform each of the modulation steps outlined in Section 5.2.1 including transmission through the PAs to estimate the receive vector y at the user. With the time-domain version of the user's data, $\mathbf{r} = \text{IFFT}(\mathbf{s})$, and their estimated time-domain receive data, $\mathbf{v}$, we learn a nonlinear model, $\hat{g}(\cdot)$, representing the effective nonlinearity in the direction of the user

vDPD Application

Any algorithm that linearizes an effective nonlinerity (vPA), we refer to as a vDPD. After learning a vPA, $\hat{g}(\cdot)$, we can estimate the received error in each subcarrier through this low-complexity proxy for the system's nonlinearities so that we may perform vDPD on this effective nonlinearity. The user data is converted to the time domain where it goes through $\hat{g}$. The estimated time-domain receive signal is then converted back to the frequency domain to get $\hat{y}$ for all subcarriers, including the guard-band subcarriers $w \in \mathcal{Z}$. The user data vector is then updated as $\mathbf{s}^{(i+1)} = \mathbf{s}^{(i)} - \mu \left(\hat{\mathbf{b}}^{(i)} - \mathbf{s}_k \right)$, where i denotes an iteration index, and μ is a learning rate.

In this work, we assume perfect channel state information (CSI). Practically, it would not be possible to directly measure CSI on the guard-band subcarriers as the users do not transmit pilots on these subcarriers. However, it may be possible to extrapolate these subcarriers by extending known CSI or applying interpolation-based techniques common for OFDM denoising [101].

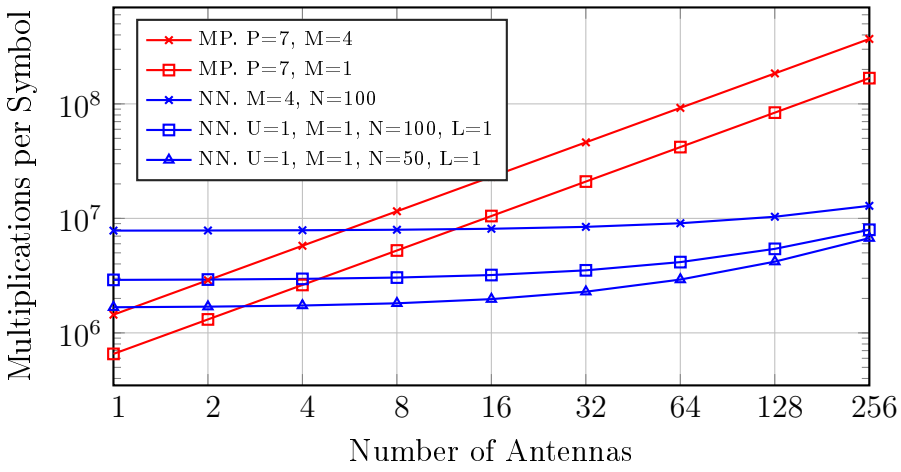


Figure 5.7: Multiplications per OFDM symbol versus the number of antennas.

Neural Network vPA Architecture

To solve for a reasonable solution to $\hat{g}$, we train a NN. We use a feed-forward MultiLayer Perceptron (MLP) NN that is fully-connected with K hidden layers, and N neurons per hidden layer. The nonlinear activation applied in hidden layers is chosen to be a ReLU, shown in (5.6), which can easily be implemented with a single multiplexer in hardware.

A model of each PA, f, is calculated. While any nonlinear model could be used, we utilize a memory polynomial (MP) as in Eq. (2.3). Then the NN engine can perform a forward pass through the system to calculate the expected error seen at the receiver. This error is then backpropagated through the system to update the NN before the precoder. MIMOSA for Python (MIMOSApy) is configured to train on a graphics processing unit (GPU) with the Adam optimizer [102] in PyTorch [87] for this training.

5.2.2 Running Complexity

Using commonplace approaches such as the MP DPD will require linearizing each PA individually. When considering the large number of antennas considered in 5G and beyond, the complexity can quickly become prohibitive. The main advantage of our proposed approach is that, while the MP-per-PA approach scales with the number of antennas, the vDPD uses a small NN. In Fig. 5.7 we plot the complexity of the MP-per-antenna DPD and the vDPD versus the number of transmit antennas N. Here, we fix the memory to $M = 4$, for all systems, and we consider the case where there are 3240 data subcarriers. The signal is upsampled to 16384 samples per OFDM symbol for both DPDs to be over 3x upsampling. The MP-per-antenna DPD, shown in red, increases linearly as each new antenna requires a new MP. The vDPD, shown in blue, requires only one NN for all cases. We consider a fixed number of hidden layers, $H = 1$. For the vDPD scheme, the complexity increases with the number of antennas due to multiplications associated with the additional precoding of guard-band subcarriers. While the running complexity may have a higher upfront cost due to the added FFT/IFFT and precoding of guard-band subcarriers, there can be a complexity advantage, without sacrificing performance, in systems where there are more than eight antennas.

5.2.3 Summary for Single-User vDPD

While massive MISO does not have the same prestige as massive MIMO, it is a practical scenario that should be considered. System coverage is a major concern in 5G systems [19], and maintaining the full array gain by limiting the number of users may be necessary in some scenarios. Many 5G arrays operate explicit beamforming with a fixed codebook where the number of users is restricted [34]. Moreover the large antenna array may be used as a point-to-point system for a fixed backhaul where there

is effectively one beam. In scenarios with 32 antennas, we can achieve approximately 3x reduction in complexity, and for 64 antennas we achieve a 5x reduction.

5.3 vDPD for Multiple Users/Many Antenna

We now present the full vDPD scheme for MU massive MIMO. Conceptually, this is similar to the scheme for MISO systems presented in the previous section. However, there is one complication in that spatial intermodulation products can occur, as shown in Eq. (4.10). In the system simulations shown in Fig. 4.13, OOB energy can be seen throughout the angular domain in directions distinct from the primary beams. While these spurious beams are frequently neglected in the literature, they too must be corrected to meet the spectral emission mask.

The main idea remains similar to the previous section. However, we now augment the design with a new concept that we call virtual user equipment (UE)s (vUEs). The main idea of vUEs is to add an virtual user input to the precoder, and corresponding columns in the precoder matrix, for each spurious beam. The vUE precoder columns are fixed with the expected angle of departure (AoD) of the beams, as calculated by Eq. (4.13). Then the NN before the precoder will use the input user data to minimize the error at the receiving users in the far-field of the array and minimize the OOB emissions in the directions of the users and vUEs.

5.3.1 System Model

We consider a fully digital, MU massive MIMO system with U single antenna users and N PAs and antenna radio frequency (RF) units at the BS. Without loss of generality, we restrict the presentation below to one OFDM symbol. A symbol of data to the users is represented by the vector $\mathbf{s}_w \in \mathcal{O}^U$, where w indexes the OFDM

tones from 1 to W and $\mathcal{O}$ represents the set of complex-valued constellation points. Pulse shaping is applied via the inclusion of guard-band subcarriers that are normally empty.

Linear precoding is applied separately to each OFDM tone, generating W vectors $\mathbf{x}_w \in \mathbb{C}^N$ with $\mathbf{x}_w = \mathbf{G}_w\mathbf{s}_w$. Here, $\mathbf{G}_w \in \mathbb{C}^{N\times U}$ is the precoding matrix such as zero-forcing (ZF) or maximum ratio transmission (MRT). Each vector is remapped to contain all the tones per antenna, $[\mathbf{x}_1, \ldots, \mathbf{x}_W] = [\mathbf{a}_1, \ldots, \mathbf{a}_N]^T$, where each $\mathbf{a}_n$ is a W-dimensional vector containing all tones for antenna port $n \in \{1, \ldots, N\}$. At this point, the data is converted from the frequency domain to the time domain via the inverse discrete Fourier transform (IDFT), which is typically calculated via an IFFT. The data is reorganized to be serial instead of parallel, and a CP is added. In many systems, windowing is also applied between symbol boundaries to improve the spectral shaping [25]. We express this time-domain representation for each antenna as the vector $\mathbf{u}_n$. This vector is upconverted to an RF frequency where it is transmitted through a PA with nonlinear function $f_n(\cdot)$. The time-domain data for each antenna is given as $\hat{\mathbf{u}}_n = f_n(\mathbf{u}_n)$, equivalently expressed as a discrete-time signal, $\hat{u}[i] = [\mathbf{u}_n]_i$. The frequency-domain equivalent is given as $\hat{\mathbf{x}}_n$.

In OFDM systems, the channel is usually modeled in the frequency-domain for each tone w as, $y_w = \mathbf{h}_w\hat{\mathbf{x}}_w + n_w$, where y_w denotes the received data for OFDM tone w and $\mathbf{h}_w$ is the $1 \times N$ channel vector, and n_w is a Gaussian random noise term. The user received signal can be remapped to $[y_1, \ldots, y_W] = \mathbf{b}$ to represent a W dimensional vector of all tones received at the user. The time-domain user-received signal is given as $\mathbf{v}$.

5.3.2 Virtual DPD NN Algorithm

The system architecture of our proposed vDPD scheme for MU-massive MIMO is illustrated in Fig. 5.8. Our method's main idea is to train a NN to predistort so that the farfield responce in the user and spurious directions is corrected.

vDPD NN Training

A block diagram of the system is presented in Fig. 5.8. The system calculates PA models for each PA using dedicated feedback paths and typical nonlinear model such as the GMP or a NN. We assume full channel information is known. We then predict the angles of the spurious beams and initialize the new columns in the precoding matrix to precode information in those directions.

For training, the user data goes through a NN predistorter, which is shown in Fig. 5.9. While the NN has an input for each user, it has outputs for each user and vUE. This data goes through the vUE enhanced precoding matrix, IFFTs, PA models, and channel matrix. The error is computed in the direction of the UEs and spurious beams. This error is then backpropagated through the system to update the NN.

The complete system can be modeled as

$$\tilde{y} = H(\hat{f}(P\hat{g}(s))). \tag{5.9}$$

We seek to minimize the farfield error, so we create a digital model of the system. We calculate a training loss to backpropagate through the system so that we may tune the NN,

$$e = \|y - \tilde{y}\|^2. \tag{5.10}$$

While the training complexity is high, it is run relatively infrequently. The PA

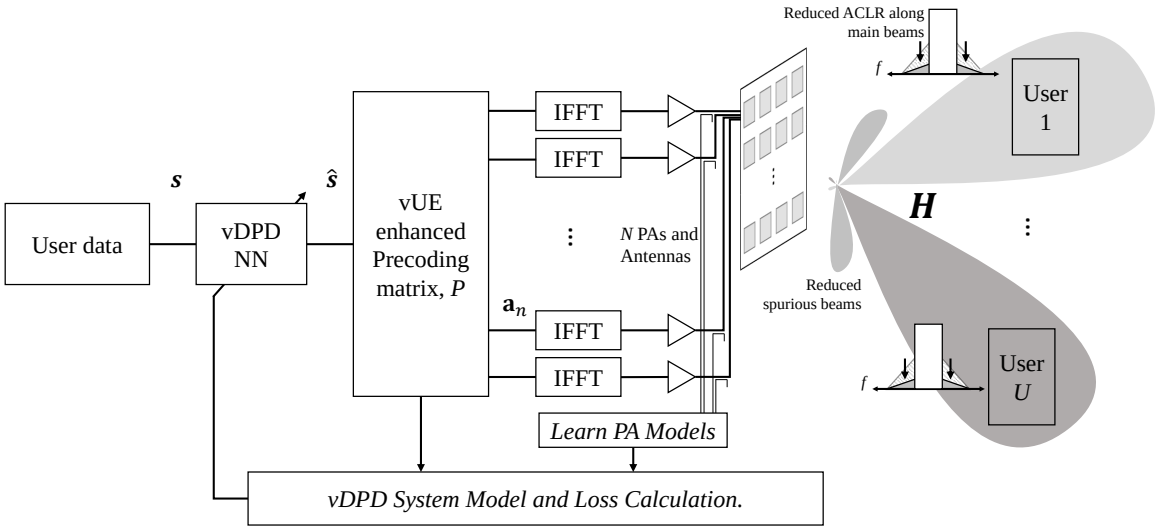


Figure 5.8: MIMO vDPD with vUEs System Block Diagram.

models that are used to create the training data do not need to be relearned often since they are based on the PAs, whose models remain relatively consistent for a given temperature. Once the NN is learned, it can be updated in a few epochs as needed to improve performance whenever needed.

5.3.3 Complexity

Using commonplace approaches such as the MP DPD will require linearizing each PA individually. When considering the large number of antennas considered in 5G and beyond, the complexity can quickly become prohibitive. The main advantage of our proposed approach is that, while the DPD-per-PA approach scales with the number of antennas, the vDPD uses a small NN. In Fig. 5.10 we plot the complexity of the GMP-per-antenna DPD and the vDPD versus the number of transmit antennas N. Here, we fix the memory to $M = 4$, for all systems, and we consider the case where there are 1200 data subcarriers. The signal is upsampled to 4096 samples per OFDM symbol for both DPDs to be over 3x upsampling. The MP-per-antenna DPD, shown in red, increases linearly as each new antenna requires a new MP. The vDPD, shown

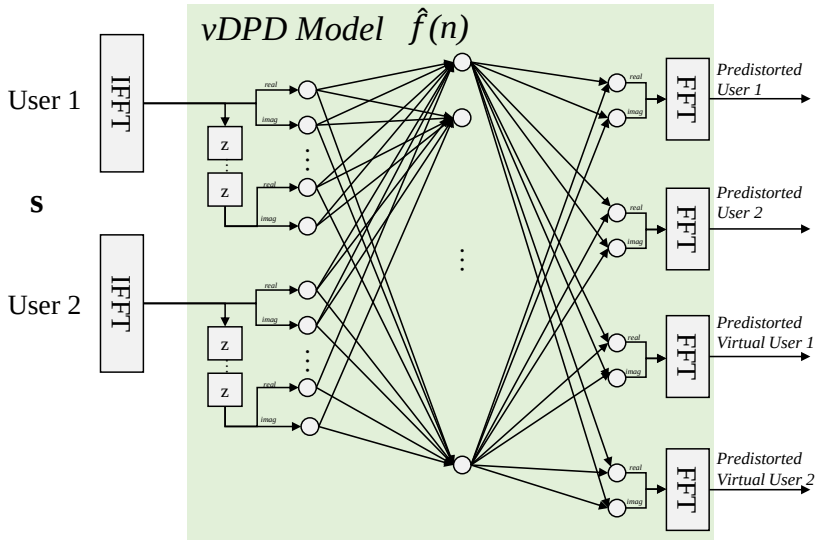


Figure 5.9: vDPD NN Block Diagram. Each user stream along with M delayed replicas are input into a NN which computes predistorted outputs for each user and vUE.

in blue, requires only one NN for all cases. Here, there is a complexity advantage to the vDPD scheme in cases where $U \leq 4$ where the ratio between antennas and users is 8:1. For example, in the case of two users and 64 antennas, the GMP scheme requires 3.8× the total number of multiplies compared to the vDPD scheme.

With the vDPD scheme with the added vUE angles, there can be a complexity advantage for some scenarios. However, as the number of users grows, the number of vUEs grows combinatorially. In Fig. 5.11, we show the growth in the number of spatial intermodulation products with the number of users. This fast growth in the number of unique intermodulation products represents a key challenge with a beam-based DPD scheme, such as ours.

5.3.4 Two-User Simulation

In this section, we show the vDPD with vUEs for two users. We consider a user at 80° and 110°. In Fig. 5.12, we show the array response in the case without vDPD. Here, the maximum ACLR is in the direction of the users at nearly -30 dBc. We

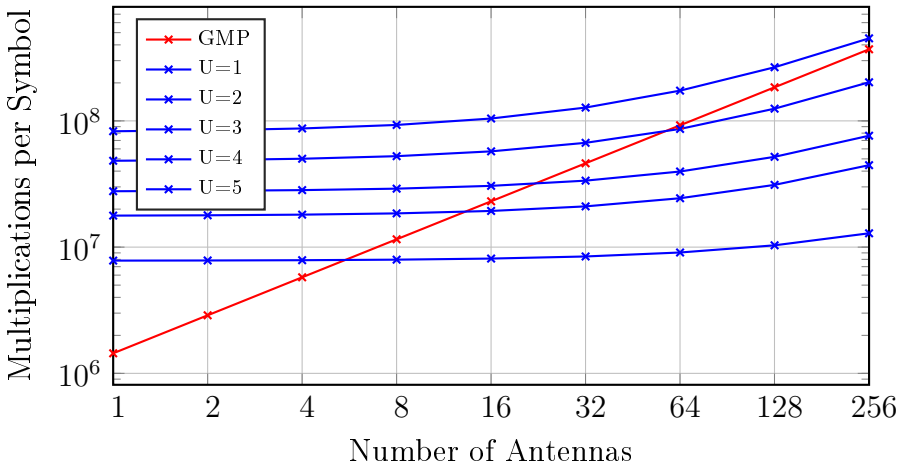


Figure 5.10: Multiplications per OFDM symbol versus the number of antennas with increasing number of users. There is a complexity advantage to the vDPD scheme in cases where $U \leq 4$ where the ratio between antennas and users is 8:1.

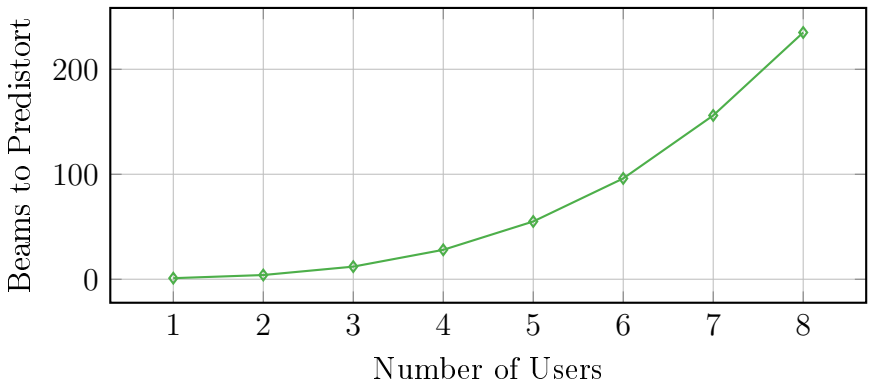


Figure 5.11: Explosion in the number of spurious beams as the number of UEs increases.

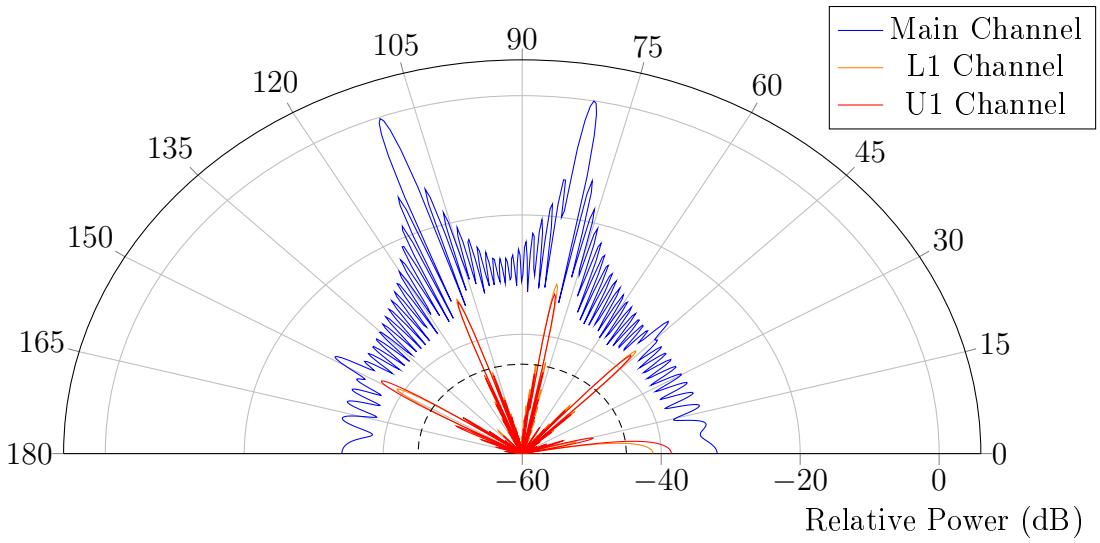


Figure 5.12: LOS example without DPD with a user at 80° and 110°.

apply the vDPD scheme with 80 neurons and one hidden layer. The final spectrum is presented in Fig. 5.13. For each vDPD result, we present the beam plot on testing symbols that were not in the set of training symbols. We show the training process in Fig. 5.14. Every 100 epochs, we create new training data to avoid overfitting.

5.3.5 Six-User Simulation

In these tests, we show the case of six users. We apply the vDPD scheme with 200 neurons and one hidden layer. In Fig. 5.15, we show the case where six users are placed at 64°, 75°, 83°, 95°, 110°, and 120°. Here, the ACLR is strongest in the direction of the users, but a peak of only -39 dBc. When compared to the case of two users, the ACLR is more evenly dispersed in the angular domain leading to lower peaks.

In Fig. 5.16, we show the case after vDPD. The ACLR is reduced in all directions with one minor violation. In this case, with the full combination of intermodulations considered, we have a total of 96 possible vUEs.

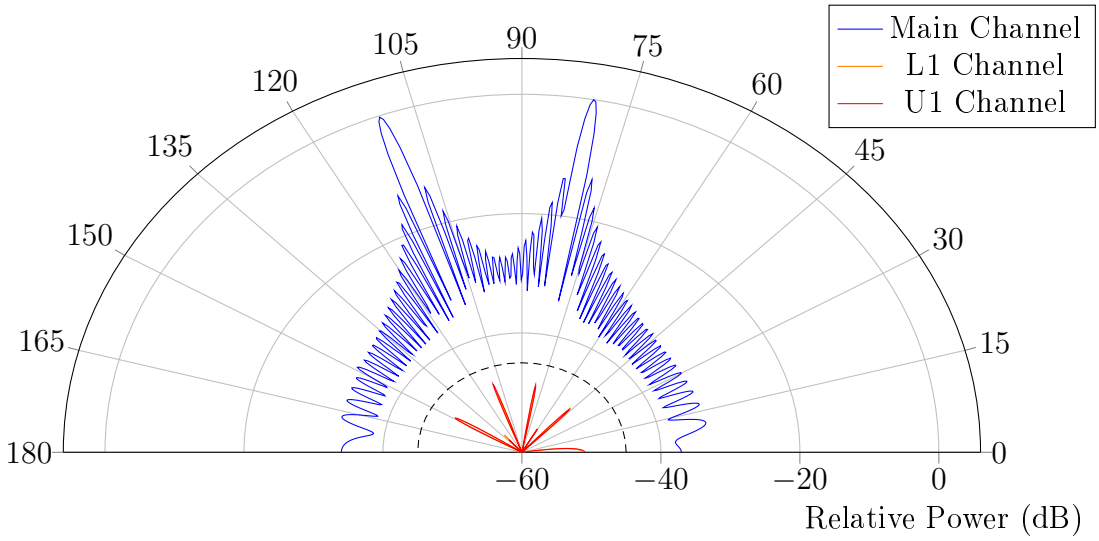


Figure 5.13: LOS example with vDPD a user at 80° and 110°.

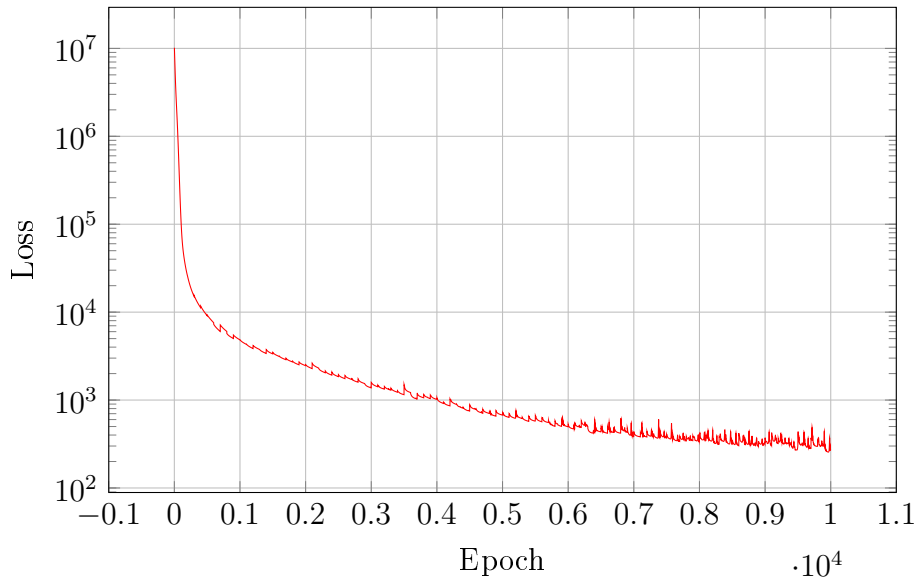


Figure 5.14: vDPD NN Training. As the NN is trained over many symbols and epochs, the total loss is reduced.

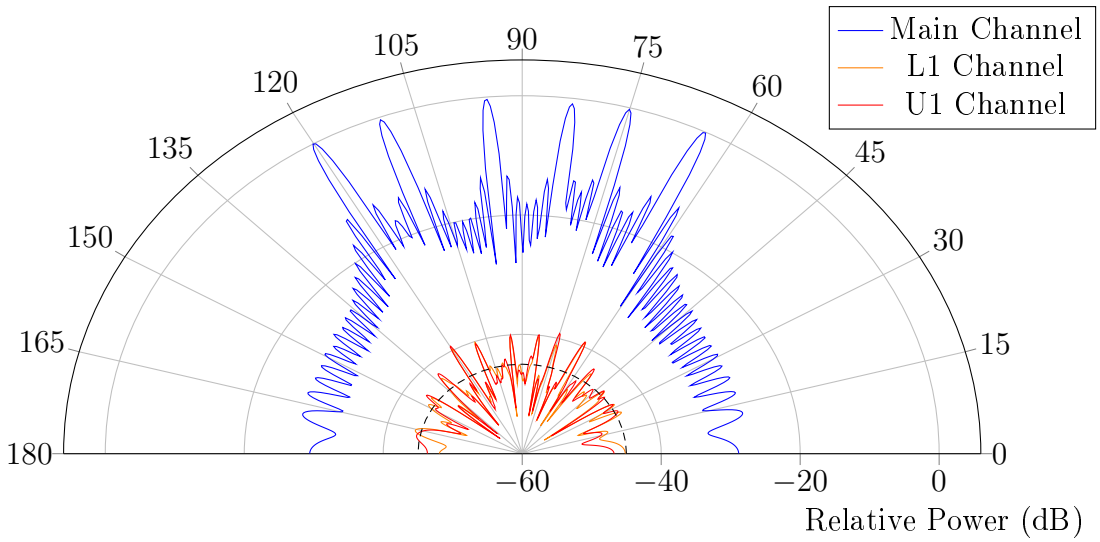


Figure 5.15: LOS beamforming to six users without DPD. In this example, there are significant OOB emissions violations throughout the angular domain.

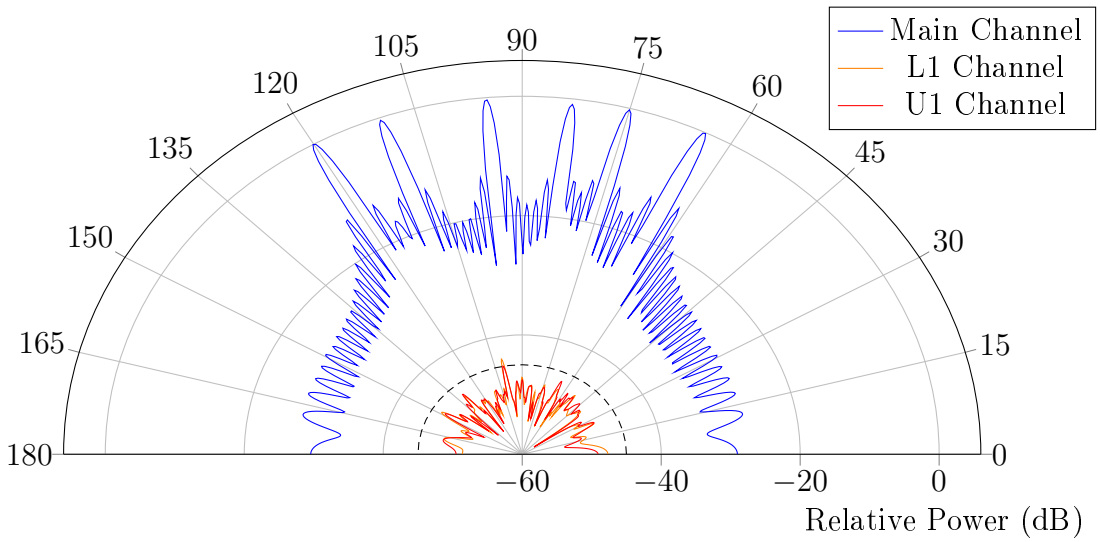


Figure 5.16: LOS beamforming to 6 users with vDPD.

5.3.6 User Mobility

In a practical deployment, users typically move throughout the environment. Due to the fact that the vDPD solution occurs before the beamforming precoder, it is natural to expect that its performance depends on the precoder. However, this is not true in many scenarios. In this section, we explore the vDPD performance while considering user mobility over a variety of scenarios by using MIMOSApy.

LoS Channels

For a line-of-sight (LoS) system, we have vDPD as follows. Assume that a user moves from position θ_{t_1} to θ_{t_2} with channel vectors $\mathbf{h}_1$ and $\mathbf{h}_2$. To mathematically illustrate the idea, consider the two-tone signal in a MIMO array with identical third-order memoryless intermodulation on each antenna, similar to Section 4.2.1,

$$y_n(k) = \left(A_1 + \frac{3\alpha_n}{2}A_1A_2^2 + \frac{3\alpha_n}{4}A_1^3\right)e^{j(\gamma_1+\phi_{1,n}(t))} \tag{5.11}$$

$$+ \left(A_2 + \frac{3\alpha_n}{2}A_1^2A_2 + \frac{3\alpha_n}{4}A_2^3\right)e^{j(\gamma_2+\phi_{2,n}(t))} \tag{5.12}$$

$$+ \frac{3\alpha_n}{4}A_1A_2^2e^{j(2\gamma_2-\gamma_1+2\phi_{1,n}(t)-\phi_{2,n}(t))} \tag{5.13}$$

$$+ \frac{3\alpha_n}{4}A_1^2A_2e^{j(2\gamma_1-\gamma_2+2\phi_{2,n}(t)-\phi_{1,n}(t))}. \tag{5.14}$$

Here, there are four beams created at array response directions $\phi_{1,n}(t)$, $\phi_{2,n(t)}$, $2\phi_{1,n}(t)-\phi_{2,n}(t)$, and$2\phi_{2,n}(t)-\phi_{1,n}(t)$. While these array response directions update with t, the generic I/Qwaveform in each array response remains the same. For example, the signal beamformed in the direction $2\phi_{1,n}(t)-\phi_{2,n}(t)$ is given as $N\frac{3\alpha}{4}A_1[k]A_2^2[k]e^{2\gamma_2[k]-\gamma_1[k]}$ which does not depend on the user positions. Hence, as long as the correct nonlinearity can be calculated and as long as the spurious direction can be predicted, the only update necessary with user mobility is to the linear precoder.

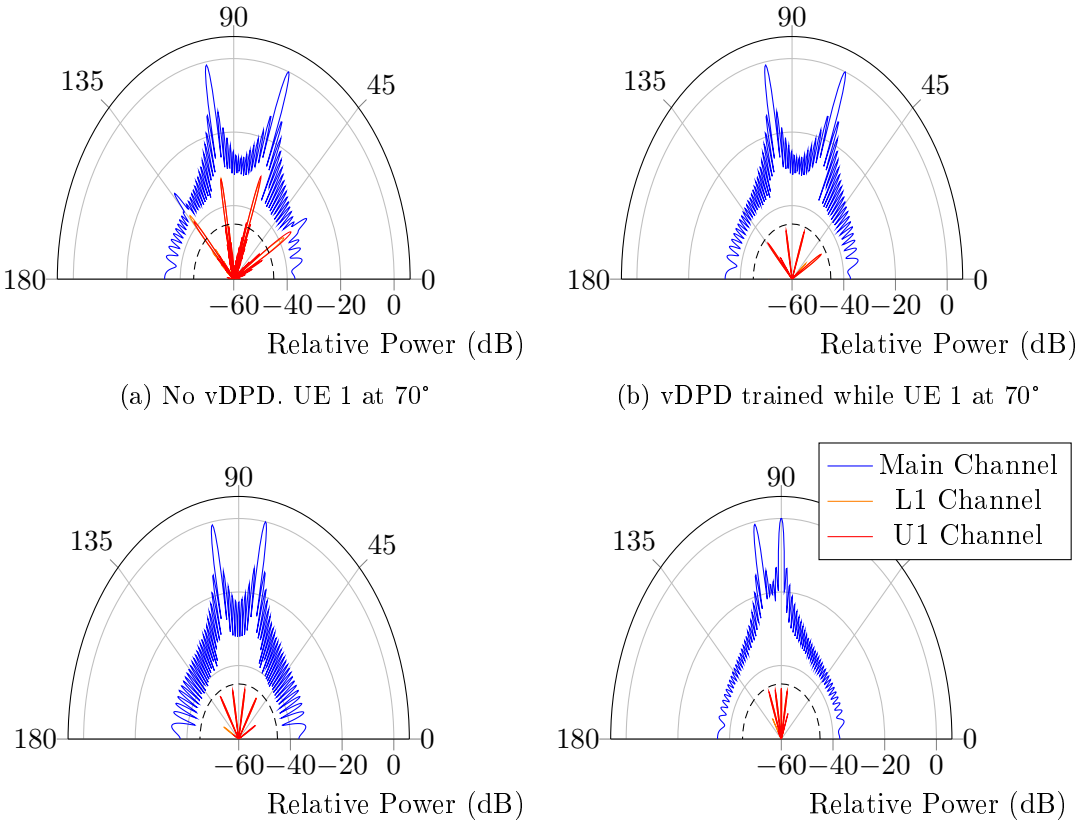
Simulations

In this section, we simulate a two user case with OFDM where we train with the users at 70° and 100°. We then step user 1 along an arc to positions of 80° and then 90°. We assume that full CSI is known by the transmitter and the precoder is updated. However, the vDPD remains fixed. We consider a uniform linear array (ULA) with 64 elements, each with third-order nonlienarities. We choose a vDPD architecture as a MLP with one hidden layer and 80 hidden neurons. We train in the first user position over 150 epochs using the Adam optimizer. We test using OFDM with 1200 data subcarriers and 4096 total subcarriers with a spacing of 15 kHz. We precode using MRT.

In Fig. 5.17, we see the result of the MIMOSApy simulation. We initially train the vDPD for user positions of 70° and 100°, shown in Fig. 5.17a and Fig. 5.17b. User 1 then moves to 80° and 100°, shown in Fig. 5.17c and Fig. 5.17d. While the precoder is updated to steer the beams in new directions, the relative contents of each beam remains constant. The signal beamformed to the spurious beams remains an intermodulation of the signals beamformed to the users and hence can be predistored by the same vDPD function.

5.4 Other Schemes Explored

While developing vDPD, many other schemes were explored. While each idea had merit, we ultimately abandoned each in favor of the MU-vDPD shown in the previous section. In this section, we provide a brief overview of a few notable schemes and their limitations. While we do not provide and document results, we hope that section may provide interesting notes for anyone pursuing similar ideas.



(a) No vDPD. UE 1 at 70°

(b) vDPD trained while UE 1 at 70°

(c) Reuse vDPD while UE 1 moves to 80°

(d) Reuse vDPD while UE 1 moves to 90°

Figure 5.17: LOS Mobility Example. The users start at 70° and 100°. We then train the vDPD for the main and spurious beams. User 1 moves from 70° to 80°, and we reuse the previously trained vDPD while updating the precoder. The user then moves from 80° to 90°without sacrificing the vDPD performance.

Linear Precoder DPD Most MIMO systems use a linear precoder to generate the symbols to transmit at each antenna. Typically, ZF or MRT is adopted to optimize signal-to-interference-plus-noise ratio (SINR) or signal-to-noise ratio (SNR). However, this is done without considering the imperfections including the PA nonlinearity. While reformulating the linear precoder solution to include nonlinearity is mathematically intractable, machine learning tools can be used to tune weights. We pursued this strategy and used MIMOSApy to learn the precoder weights after initializing them to the ZF solution. However, it became quickly apparent that there is no mechanism for a linear precoder to apply predistortion so that the OOB emissions could be reduced. While some in-band error vector magnitude (EVM) could be improved, the linear precoder could have thousands of parameters and hence was vulnerable to over-fitting and long training times with no notable advantage. We also performed initial tests where we learned the precoder along with the vDPD NN. However, this too provided no advantage when compared to vDPD with the standard precoder solutions.

Neural Nonlinear Precoder Nonlinear precoding schemes such as dirty paper coding (DPC) are able to maintain the capacity as if the channel were free of interference [103]. One idea we considered was to develop a joint vDPD precoding scheme that would correct for nonlinearities and precode for the antennas via one NN. Similar to above, this drastically increased the size of our NN making it difficult to train and converge. By separating the predistortion functionality and the precoding functionality, we are able to keep the NN relatively small for fast training. Long training times from NN-based precoders ultimately may make it challenging to compute updates within the channel coherence time.

Multi-User vPA In this thesis, we have considered a beam-based pridistortion method where we correct the nonlinearities in each beam, without explicitly modeling the effective nonlinearities of vPAs. We originally began the project by explicitly modeling the nonlinearity to each user as if there was one effective nonlinearity in that direction. Based on this vPA, we could then use an ILA or other traditional DPD training technique to train some sort of predistorter before the precoder corresponding to that beam. However, in multi-user scenarios, we found that beam intermodulation could occur, creating additional spurious beams. Considering spurious beams in this vPA-focused scheme became difficult as we could not directly create a vPA corresponding to this spurious beam where there was no intended signal. We then pivoted our focus to concentrate on the predistortion side, where we would consider the total error in multiple directions leading us to the vDPD scheme focused on in this thesis.

vDPD-based on precoder pseudoinverse In [22], we considered a vDPD scheme where we learned the MP-based predistorter for each antenna to generate ideal, predistorted data. We then multiplied this predistorted data by the pseudoinverse of the precoder matrix, where we trained a NN similar to the NN in this thesis. While this scheme worked well for the primary, user directions, there was limited performance in the spurious directions.

Chapter 6

Conclusions

To enable the next generation of cheaper, more efficient massive multiple-input, multiple-output (MIMO) technologies it is critical to reduce the complexity throughout the processing chain. In this work, we show a novel method of predistorting to correct for the nonlinear amplifiers. In particular, we show that it is possible to perform this linearization before the precoding beamformer, even when there are multiple users. By utilizing a neural network, we are able to reduce the complexity by up to a factor of 4x in cases with a high number of antennas and low number of users.

We have presented in-depth simulations exploring the problem with a variety of complex impairments in the power amplifier (PA) models including memory effects and high order nonlinearity. We have used an actual Doherty PA array to establish the baseline model and show the expected behavior. We have also studied the effect on the out-of-band (OOB) energy when beamforming. We found that the OOB energy is dominant in the direction of the intended beamforming. In multi-user cases, secondary beams appear separate from the main beam due to intermodulation of the user data. We have developed a scheme to correct for these nonlinearites before the precoder using a neural network.

6.1 Possibilities for Future Exploration

6.1.1 Experimental Verification

There were multiple avenues of experimental verification that were pursued. In particular, we worked to perform verification with Reconfigurable Eco-system for Next-generation End-to-end Wireless (RENEW). However, there are many challenges when it comes to working with hardware platforms. We built MIMOSA for Python (MIMOSApy) in Section 3.2, a machine-learning-enabled interface to RENEW for rapid training and testing with RENEW and used it to measure the beamforming response in Section 4.1.7. However, we were unable to complete the virtual digital predistortion (DPD) (vDPD) verification due to unforeseen issues with the frontend hardware. While RENEW has the potential to provide insights due to the large number of antennas, it is still significantly limited in the usable bandwidth, forcing us to utilize signals with channel bandwidths as low as 1.4 MHz whereas the industry is currently increasing bandwidth as high as 100 MHz in similar spectrum bands. The wideband PA testbed from Section 4.1.1 provides an alternative platform to perform testing with 5G new radio (NR) signals of 100 MHz. However, with only 16 antennas, the possible number of users to test remains low.

6.1.2 Additional Optimizations

Currently there are multiple avenues of potential improvements to be made. In particular, the primary challenge is addressing the case of many users in massive MIMO. We are interested in exploring the case of allowing the neural network (NN) to augment the precoder to help add degrees of freedom to reduce the adjacent channel leakage ratio (ACLR).

6.1.3 Additional Investigations

The vDPD scheme is an entirely new method of performing predistortion in wireless communications. By moving the predistortion up the signal processing chain, there are many possible interaction that can occur. In particular, the relationship between the vDPD and the precoder has not been fully explored. Considerations for how often the vDPD scheme needs to be updated with respect to changes in the channel and precoder are left for future work. Considerations for real-time training are also left for future work.

6.2 Impact

This scheme can have impacts throughout the industry. For example, many practical massive MIMO systems currently use explicit beamforming with fixed codebooks of beams. Moreover, the number of downlink streams is often only one to two users, where we showed our system to have the most complexity reduction. Any large-scale MIMO communications system with fixed beams such as 5G-to-the-home or other backhaul systems or line-of-sight (LoS) MIMO [47] may benefit from the vDPD schemes outlined in this thesis. In these cases, the NN vDPD solution could be deployed for significant computational savings and only minor retraining over time.

Bibliography

[1] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, 2014. DOI: 10.1109/MCOM.2014.6736761.

[2] C. Desset, B. Debaillie, V. Giannini, *et al.*, "Flexible power modeling of LTE base stations," in *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, 2012, pp. 2858–2862. DOI: 10.1109/WCNC.2012.6214289.

[3] L. Guan and A. Zhu, "Green communications: Digital predistortion for wide-band RF power amplifiers," *IEEE Microwave Magazine*, vol. 15, no. 7, pp. 84–99, 2014. DOI: 10.1109/MMM.2014.2356037.

[4] W. Chen, G. Lv, X. Liu, D. Wang, and F. M. Ghannouchi, "Doherty PAs for 5G massive MIMO: Energy-efficient integrated DPA MMICs for sub-6-GHz and mm-wave 5G massive MIMO systems," *IEEE Microwave Magazine*, vol. 21, no. 5, pp. 78–93, 2020. DOI: 10.1109/MMM.2020.2971183.

[5] F. M. Ghannouchi and O. Hammi, "Behavioral modeling and predistortion," *IEEE Microwave Magazine*, vol. 10, no. 7, pp. 52–64, 2009. DOI: 10.1109/MMM.2009.934516.

[6] A. Katz, J. Wood, and D. Chokola, "The evolution of PA linearization: From classic feedforward and feedback through analog and digital predistortion," *IEEE Microwave Magazine*, vol. 17, no. 2, pp. 32–40, 2016. DOI: 10.1109/MMM.2015.2498079.

[7] M. Abdelaziz, C. Tarver, K. Li, *et al.*, "Sub-band digital predistortion for noncontiguous transmissions: Algorithm development and real-time prototype implementation," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, 2015, pp. 1180–1186. DOI: 10.1109/ACSSC.2015.7421326.

[8] M. Abdelaziz, L. Anttila, C. Tarver, K. Li, J. R. Cavallaro, and M. Valkama, "Low-complexity subband digital predistortion for spurious emission suppression in noncontiguous spectrum access," *IEEE Transactions on Microwave Theory and Techniques*, vol. 64, no. 11, pp. 3501–3517, 2016. DOI: 10.1109/TMTT.2016.2602208.

[9] C. Tarver, M. Abdelaziz, L. Anttila, M. Valkama, and J. R. Cavallaro, "Low-complexity, sub-band DPD with sequential learning: Novel algorithms and WARPLab implementation," in *2016 IEEE International Workshop on Signal Processing Systems (SiPS)*, 2016, pp. 303–308. DOI: 10.1109/SiPS.2016.60.

[10] C. Tarver, M. Abdelaziz, L. Anttila, and J. R. Cavallaro, "Multi component carrier, sub-band DPD and GNURadio implementation," in *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2017, pp. 1–4. DOI: 10.1109/ISCAS.2017.8050455.

[11] K. Li, A. Ghazi, C. Tarver, *et al.*, "Parallel digital predistortion design on mobile GPU and embedded multicore CPU for mobile transmitters," *J. Signal Process. Syst.*, vol. 89, no. 3, pp. 417–430, 2017.

[12] C. Tarver, A. Balatsoukas-Stimming, and J. R. Cavallaro, "Design and implementation of a neural network based predistorter for enhanced mobile broadband," in *2019 IEEE International Workshop on Signal Processing Systems (SiPS)*, 2019, pp. 296–301. DOI: 10.1109/SiPS47522.2019.9020606.

[13] C. Tarver, L. Jiang, A. Sefidi, and J. R. Cavallaro, "Neural network DPD via backpropagation through a neural network model of the PA," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 358–362. DOI: 10.1109/IEEECONF44664.2019.9048910.

[14] C. Tarver, A. Balatsoukas-Stimming, and J. R. Cavallaro, "Predistortion of OFDM waveforms using guard-band subcarriers," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*, 2020, pp. 12–16. DOI: 10.1109/IEEECONF51394.2020.9443468.

[15] C. Tarver, A. Singhal, and J. R. Cavallaro, "GPU-based linearization of MIMO arrays," in *2020 IEEE Workshop on Signal Processing Systems (SiPS)*, 2020, pp. 1–5. DOI: 10.1109/SiPS50750.2020.9195239.

[16] C. Tarver, A. Balatsoukas-Stimming, C. Studer, and J. R. Cavallaro, "OFDM-based beam-oriented digital predistortion for massive MIMO," in *IEEE Int. Sym. on Circuits and Systems*, 2021, pp. 1–5. DOI: 10.1109/ISCAS51556.2021.9401479.

[17] C. Tarver, M. Tonnemacher, H. Chen, J. Zhang, and J. R. Cavallaro, "GPU-based, LDPC decoding for 5G and beyond," *IEEE Open Journal of Circuits and Systems*, vol. 2, pp. 278–290, 2021. DOI: 10.1109/OJCAS.2020.3042448.

[18] K. Li, J. McNaney, C. Tarver, *et al.*, "Design trade-offs for decentralized baseband processing in massive MU-MIMO systems," in *2019 53rd Asilomar Con-*

ference on Signals, Systems, and Computers, 2019, pp. 906–912. DOI: 10.1109/IEEECONF44664.2019.9048727.

[19] H. Ji, Y. Kim, K. Muhammad, *et al.*, "Extending 5G TDD coverage with XDD: Cross division duplex," *IEEE Access*, vol. 9, pp. 51 380–51 392, 2021. DOI: 10.1109/ACCESS.2021.3068977.

[20] C. Tarver, M. Tonnemacher, V. Chandrasekhar, *et al.*, "Enabling a "Use-or-Share" Framework for PAL–GAA Sharing in CBRS Networks via Reinforcement Learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 716–729, 2019. DOI: 10.1109/TCCN.2019.2929147.

[21] M. Tonnemacher, C. Tarver, J. Cavallar, and J. Camp, "Machine learning enhanced channel selection for unlicensed LTE," in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–10. DOI: 10.1109/DySPAN.2019.8935859.

[22] C. Tarver, A. Balalsoukas-Slimining, C. Studer, and J. R. Cavallaro, "Virtual DPD neural network predistortion for OFDM-based MU-Massive MIMO," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, 2021, pp. 376–380. DOI: 10.1109/IEEECONF53345.2021.9723343.

[23] J. Salz, "Digital transmission over cross-coupled linear channels," *AT&T Technical Journal*, vol. 64, no. 6, pp. 1147–1159, 1985. DOI: 10.1002/j.1538-7305.1985.tb00269.x.

[24] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3590–3600, 2010. DOI: 10.1109/TWC.2010.092810.091092.

[25] R. Van Nee and R. Prasad, *OFDM for Wireless Multimedia Communications*, 1st. USA: Artech House, Inc., 2000, ISBN: 0890065306.

[26] C. Shepard, H. Yu, N. Anand, *et al.*, "Argos: Practical many-antenna base stations," in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, ser. Mobicom '12, Istanbul, Turkey: Association for Computing Machinery, 2012, pp. 53–64, ISBN: 9781450311595. DOI: 10.1145/2348543.2348553. [Online]. Available: https://doi.org/10.1145/2348543.2348553.

[27] C. Shepard, J. Ding, R. E. Guerra, and L. Zhong, "Understanding real many-antenna MU-MIMO channels," in *2016 50th Asilomar Conference on Signals, Systems and Computers*, 2016, pp. 461–467. DOI: 10.1109/ACSSC.2016.7869082.

[28] E. Everett, C. Shepard, L. Zhong, and A. Sabharwal, "Softnull: Many-antenna full-duplex wireless via digital beamforming," *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8077–8092, 2016. DOI: 10.1109/TWC.2016.2612625.

[29] J. Breen, A. Buffmire, J. Duerig, *et al.*, "POWDER: Platform for open wireless data-driven experimental research," in *Proceedings of the 14th International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH)*, Sep. 2020. DOI: 10.1145/3411276.3412204. [Online]. Available: https://www.flux.utah.edu/paper/breen-wintech20.

[30] C. Shepard, J. Blum, R. E. Guerra, R. Doost-Mohammady, and L. Zhong, "Design and implementation of scalable massive-MIMO networks," in *Proceedings of the 1st International Workshop on Open Software Defined Wireless Networks*, ser. OpenWireless'20, Toronto, ON, Canada: Association for Computing Machinery, 2020, pp. 7–13, ISBN: 9781450380119. DOI: 10.1145/3396865.3398691. [Online]. Available: https://doi.org/10.1145/3396865.3398691.

[31] J. Vieira, S. Malkowsky, K. Nieman, *et al.*, "A flexible 100-antenna testbed for massive mimo," in *2014 IEEE Globecom Workshops (GC Wkshps)*, 2014, pp. 287–293. DOI: 10.1109/GLOCOMW.2014.7063446.

[32] S. Malkowsky, J. Vieira, L. Liu, *et al.*, "The world's first real-time testbed for massive MIMO: Design, implementation, and validation," *IEEE Access*, vol. 5, pp. 9073–9088, 2017. DOI: 10.1109/ACCESS.2017.2705561.

[33] C. Zhang, *Massive FD-MIMO technology is proven in the field – will distributed FD-MIMO be next?* https://www.samsung.com/global/business/networks/insights/blog/massive-fd-mimo-technology-is-proven-in-the-field-will-distributed-fd-mimo-be-next/, (Accessed on 11/14/2020), Nov. 2020.

[34] "Massive MIMO for New Radio," Samsung, Tech. Rep., Dec. 2020. [Online]. Available: https://www.samsung.com/global/business/networks/insights/white-papers/1208-massive-mimo-for-new-radio/.

[35] W. Doherty, "A new high efficiency power amplifier for modulated waves," *Proceedings of the Institute of Radio Engineers*, vol. 24, no. 9, pp. 1163–1182, 1936. DOI: 10.1109/JRPROC.1936.228468.

[36] A. Saleh, "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Transactions on Communications*, vol. 29, no. 11, pp. 1715–1720, 1981. DOI: 10.1109/TCOM.1981.1094911.

[37] M. Yao, M. Sohul, R. Nealy, V. Marojevic, and J. Reed, "A digital predistortion scheme exploiting degrees-of-freedom for massive MIMO systems," in *2018 IEEE International Conference on Communications (ICC)*, 2018, pp. 1–5. DOI: 10.1109/ICC.2018.8422266.

[38] M. M. Shammasi and S. M. Safavi, "Performance of a predistorter based on Saleh model for OFDM systems in HPA nonlinearity," in *2012 14th International Conference on Advanced Communication Technology (ICACT)*, 2012, pp. 148–152.

[39] A. A. M. Saleh and J. Salz, "Adaptive linearization of power amplifiers in digital radio systems," *The Bell System Technical Journal*, vol. 62, no. 4, pp. 1019–1033, 1983. DOI: 10.1002/j.1538-7305.1983.tb03113.x.

[40] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. USA: Krieger Publishing Co., Inc., 1980, ISBN: 1575242834.

[41] L. Ding, G. Zhou, D. Morgan, *et al.*, "A robust digital baseband predistorter constructed using memory polynomials," *IEEE Transactions on Communications*, vol. 52, no. 1, pp. 159–165, 2004. DOI: 10.1109/TCOMM.2003.822188.

[42] D. Morgan, Z. Ma, J. Kim, M. Zierdt, and J. Pastalan, "A generalized memory polynomial model for digital predistortion of RF power amplifiers," *IEEE Trans. on Signal Process.*, vol. 54, no. 10, pp. 3852–3860, 2006. DOI: 10.1109/TSP.2006.879264.

[43] J. Kim and K. Konstantinou, "Digital predistortion of wideband signals based on power amplifier model with memory," English, *Electronics Letters*, vol. 37, no. 23, pp. 1–2, Nov. 2001.

[44] L. Anttila, P. Handel, and M. Valkama, "Joint mitigation of power amplifier and I/Q modulator impairments in broadband direct-conversion transmitters," *IEEE Trans. Microw. Theory Techn.*, vol. 58, no. 4, pp. 730–739, Apr. 2010, ISSN: 0018-9480. DOI: 10.1109/TMTT.2010.2041579.

[45] N. Benvenuto, F. Piazza, and A. Uncini, "A neural network approach to data predistortion with memory in digital radio systems," in *Proceedings of ICC '93*

- *IEEE International Conference on Communications*, vol. 1, 1993, 232–236 vol.1. DOI: 10.1109/ICC.1993.397263.

[46] P. Jaraut, M. Rawat, and F. M. Ghannouchi, "Composite neural network digital predistortion model for joint mitigation of crosstalk, I/Q imbalance, nonlinearity in MIMO transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 66, no. 11, pp. 5011–5020, 2018. DOI: 10.1109/TMTT.2018.2869602.

[47] T. Kobal, Y. Li, X. Wang, and A. Zhu, "Digital predistortion of RF power amplifiers with phase-gated recurrent neural networks," *IEEE Transactions on Microwave Theory and Techniques*, pp. 1–1, 2022. DOI: 10.1109/TMTT.2022.3161024.

[48] D. Phartiyal and M. Rawat, "LSTM-deep neural networks based predistortion linearizer for high power amplifiers," in *2019 National Conference on Communications (NCC)*, 2019, pp. 1–5. DOI: 10.1109/NCC.2019.8732178.

[49] Z. Liu, X. Hu, L. Xu, W. Wang, and F. M. Ghannouchi, "Low computational complexity digital predistortion based on convolutional neural network for wideband power amplifiers," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 3, pp. 1702–1706, 2022. DOI: 10.1109/TCSII.2021.3109973.

[50] Y. Wu, U. Gustavsson, M. Valkama, A. G. i. Amat, and H. Wymeersch, *Frequency-domain digital predistortion for massive MU-MIMO-OFDM downlink*, 2022. DOI: 10.48550/ARXIV.2205.05158. [Online]. Available: https://arxiv.org/abs/2205.05158.

[51] D. Blalock, J. J. G. Ortiz, J. Frankle, and J. Guttag, *What is the state of neural network pruning?* 2020. DOI: 10.48550/ARXIV.2003.03033. [Online]. Available: https://arxiv.org/abs/2003.03033.

[52] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, *A survey of quantization methods for efficient neural network inference*, 2021. DOI: 10.48550/ARXIV.2103.13630. [Online]. Available: https://arxiv.org/abs/2103.13630.

[53] F. A. Aoudia and J. Hoydis, "End-to-end learning of communications systems without a channel model," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, 2018, pp. 298–303. DOI: 10.1109/ACSSC.2018.8645416.

[54] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Model-driven deep learning for MIMO detection," *IEEE Transactions on Signal Processing*, vol. 68, pp. 1702–1715, 2020. DOI: 10.1109/TSP.2020.2976585.

[55] A. Balatsoukas-Stimming and C. Studer, "Deep unfolding for communications systems: A survey and some new directions," in *2019 IEEE International Workshop on Signal Processing Systems (SiPS)*, 2019, pp. 266–271. DOI: 10.1109/SiPS47522.2019.9020494.

[56] Changsoo Eun and E. J. Powers, "A new Volterra predistorter based on the indirect learning architecture," *IEEE Transactions on Signal Processing*, vol. 45, no. 1, pp. 223–227, 1997. DOI: 10.1109/78.552219.

[57] D. Psaltis, A. Sideris, and A. Yamamura, "A multilayered neural network controller," *IEEE Control Systems Magazine*, vol. 8, no. 2, pp. 17–21, 1988. DOI: 10.1109/37.1868.

[58] J. Chani-Cahuana, P. N. Landin, C. Fager, and T. Eriksson, "Iterative learning control for RF power amplifier linearization," *IEEE Transactions on Mi-*

crowave Theory and Techniques, vol. 64, no. 9, pp. 2778–2789, 2016. DOI: 10.1109/TMTT.2016.2588483.

[59] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986. DOI: 10.1038/323533a0. [Online]. Available: http://www.nature.com/articles/323533a0.

[60] A. T. Kristensen, A. Burg, and A. Balatsoukas-Stimming, "Identification of non-linear RF systems using backpropagation," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2020, pp. 1–6. DOI: 10.1109/ICCWorkshops49005.2020.9145367.

[61] R. N. Braithwaite, "A comparison of indirect learning and closed loop estimators used in digital predistortion of power amplifiers," in *2015 IEEE MTT-S International Microwave Symposium*, 2015, pp. 1–4. DOI: 10.1109/MWSYM.2015.7166826.

[62] J. Cavers, "Amplifier linearization using a digital predistorter with fast adaptation and low memory requirements," *IEEE Transactions on Vehicular Technology*, vol. 39, no. 4, pp. 374–382, 1990. DOI: 10.1109/25.61359.

[63] L. Guan and A. Zhu, "Low-cost FPGA implementation of Volterra series-based digital predistorter for RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 4, pp. 866–872, 2010. DOI: 10.1109/TMTT.2010.2041588.

[64] P. Campo, V. Lampu, A. Meirhaeghe, J. Boutellier, L. Anttila, and M. Valkama, "Digital predistortion for 5G small cell: GPU implementation and RF measurements," *J. Signal Process. Syst.*, vol. 92, pp. 475–486, 2020.

[65] K. Li, "Decentralized baseband processing for massive MU-MIMO systems," Ph.D. dissertation, Dept. of Elect. and Comput. Eng., Rice University, 2019. [Online]. Available: https://scholarship.rice.edu/handle/1911/105428?show=full.

[66] W. Sandrin, "Spatial distribution of intermodulation products in active phased array antennas," *IEEE Transactions on Antennas and Propagation*, vol. 21, no. 6, pp. 864–868, 1973. DOI: 10.1109/TAP.1973.1140612.

[67] C. Studer, M. Wenk, and A. Burg, "MIMO transmission with residual transmit-RF impairments," in *2010 International ITG Workshop on Smart Antennas (WSA)*, 2010, pp. 189–196. DOI: 10.1109/WSA.2010.5456453.

[68] U. Gustavsson, C. Sanchéz-Perez, T. Eriksson, *et al.*, "On the impact of hardware impairments on massive mimo," in *2014 IEEE Globecom Workshops (GC Wkshps)*, 2014, pp. 294–300. DOI: 10.1109/GLOCOMW.2014.7063447.

[69] E. Björnson, J. Hoydis, M. Kountouris, and M. Debbah, "Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits," *IEEE Transactions on Information Theory*, vol. 60, no. 11, pp. 7112–7139, 2014. DOI: 10.1109/TIT.2014.2354403.

[70] E. G. Larsson and L. Van Der Perre, "Out-of-band radiation from antenna arrays clarified," *IEEE Wireless Communications Letters*, vol. 7, no. 4, pp. 610–613, 2018. DOI: 10.1109/LWC.2018.2802519.

[71] C. Studer and E. G. Larsson, "PAR-aware large-scale multi-user MIMO-OFDM downlink," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 2, pp. 303–313, 2013. DOI: 10.1109/JSAC.2013.130217.

[72] S. Taner and C. Studer, "$\ell^p - \ell^q$-norm minimization for joint precoding and peak-to-average-power ratio reduction," 2021. arXiv: 2107.06986 [cs.IT].

[73] A. F. Molisch, V. V. Ratnam, S. Han, *et al.*, "Hybrid beamforming for massive MIMO: A survey," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 134–141, 2017. DOI: 10.1109/MCOM.2017.1600400.

[74] S. A. Bassam, M. Helaoui, S. Boumaiza, and F. M. Ghannouchi, "Experimental study of the effects of RF front-end imperfection on MIMO transmitter performance," in *2008 IEEE MTT-S International Microwave Symposium Digest*, 2008, pp. 1187–1190. DOI: 10.1109/MWSYM.2008.4633270.

[75] S. A. Bassam, M. Helaoui, and F. M. Ghannouchi, "Crossover digital predistorter for the compensation of crosstalk and nonlinearity in MIMO transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 57, no. 5, pp. 1119–1128, 2009. DOI: 10.1109/TMTT.2009.2017258.

[76] S. Amin, P. N. Landin, P. Händel, and D. Rönnow, "Behavioral modeling and linearization of crosstalk and memory effects in RF MIMO transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 62, no. 4, pp. 810–823, 2014. DOI: 10.1109/TMTT.2014.2309932.

[77] *Zynq UltraScale+ RFSoC Data Sheet: DC and AC Switching Characteristics*, DS926, v1.8, Xilinx, Apr. 2021.

[78] A. Brihuega, L. Anttila, and M. Valkama, "Neural-network-based digital predistortion for active antenna arrays under load modulation," *IEEE Microwave and Wireless Components Letters*, vol. 30, no. 8, pp. 843–846, 2020. DOI: 10.1109/LMWC.2020.3004003.

[79] X. Liu, Q. Zhang, W. Chen, *et al.*, "Beam-oriented digital predistortion for 5G massive MIMO hybrid beamforming transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 66, no. 7, pp. 3419–3432, 2018. DOI: 10.1109/TMTT.2018.2830772.

[80] S. R. Aghdam, S. Jacobsson, U. Gustavsson, G. Durisi, C. Studer, and T. Eriksson, "Distortion-aware linear precoding for massive MIMO downlink systems with nonlinear power amplifiers," *CoRR*, vol. abs/2012.13337, 2020. arXiv: 2012.13337. [Online]. Available: https://arxiv.org/abs/2012.13337.

[81] M. Abdelaziz, L. Anttila, and M. Valkama, "Reduced-complexity digital predistortion for massive MIMO," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 6478–6482. DOI: 10.1109/ICASSP.2017.7953404.

[82] C. Yu, J. Jing, H. Shao, *et al.*, "Full-angle digital predistortion of 5G millimeter-wave massive MIMO transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 67, no. 7, pp. 2847–2860, 2019. DOI: 10.1109/TMTT.2019.2918450.

[83] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y.-J. A. Zhang, "The roadmap to 6G: AI empowered wireless networks," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 84–90, 2019. DOI: 10.1109/MCOM.2019.1900271.

[84] C. Tarver, *MIMOSA: MIMO Simulator with Amplifiers*, Jan. 2022. DOI: 10.5281/zenodo.5898265.

[85] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. USA: Addison-Wesley Longman Publishing Co., Inc., 1995, ISBN: 0201633612.

[86] F. Chollet *et al.*, *Keras*, https://keras.io, 2015.

[87] A. Paszke, S. Gross, F. Massa, *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., Curran Associates, Inc., 2019, pp. 8024–8035.

[Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[88] Xilinx, "Zynq ultrascale+ RFSoC ZCU208 evaluation kit," [Online]. Available: https://www.xilinx.com/publications/product-briefs/xilinx-zcu208-product-brief.pdf.

[89] *NXP Semiconductors RapidRF front-end design*, NXP Semiconductors, 2021. [Online]. Available: https://www.nxp.com/products/radio-frequency/rf-power/rf-cellular-infrastructure/rapidrf-front-end-designs:RAPIDRF-FRONTEND.

[90] *Power amplifier module for LTE and 5G*, AFSC5G35D35, Rev. 2, NXP Semiconductors, May 2019. [Online]. Available: https://www.nxp.com/docs/en/data-sheet/AFSC5G35D35.pdf.

[91] S. Jaeckel, L. Raschkowski, K. Börner, and L. Thiele, "Quadriga: A 3-D multi-cell channel model with time evolution for enabling virtual field trials," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 6, pp. 3242–3256, 2014. DOI: 10.1109/TAP.2014.2310220.

[92] J. Ding, R. Doost-Mohammady, A. Kalia, and L. Zhong, "Agora: Real-time massive mimo baseband processing in software," in *Proceedings of the 16th International Conference on Emerging Networking EXperiments and Technologies*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 232–244, ISBN: 9781450379489. [Online]. Available: https://doi.org/10.1145/3386367.3431296.

[93] C. Mollén, U. Gustavsson, T. Eriksson, and E. G. Larsson, "Spatial characteristics of distortion radiated from antenna arrays with transceiver nonlinearities,"

IEEE Trans. on Wireless Commun., vol. 17, no. 10, pp. 6663–6679, 2018. DOI: 10.1109/TWC.2018.2861872.

[94] M. Abdelaziz, L. Anttila, A. Brihuega, F. Tufvesson, and M. Valkama, "Digital predistortion for hybrid MIMO transmitters," *IEEE J. of Sel. Topics Signal Process.*, vol. 12, no. 3, pp. 445–454, 2018. DOI: 10.1109/JSTSP.2018.2824981.

[95] L. Anttila, A. Brihuega, and M. Valkama, "On antenna array out-of-band emissions," *IEEE Wireless Communications Letters*, vol. 8, no. 6, pp. 1653–1656, 2019. DOI: 10.1109/LWC.2019.2934442.

[96] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge, U.K.: Cambridge University Press, 2005, ISBN: 0521845270.

[97] C. Hemmi, "Pattern characteristics of harmonic and intermodulation products in broadband active transmit arrays," *IEEE Transactions on Antennas and Propagation*, vol. 50, no. 6, pp. 858–865, 2002. DOI: 10.1109/TAP.2002.1017668.

[98] C. Hsu and H. Liao, "PAPR reduction using the combination of precoding and mu-law companding techniques for OFDM systems," in *IEEE 11th Int. Conf. on Signal Process.*, vol. 1, 2012, pp. 1–4.

[99] S. Brandes, I. Cosovic, and M. Schnell, "Reduction of out-of-band radiation in OFDM systems by insertion of cancellation carriers," *IEEE Communications Letters*, vol. 10, no. 6, pp. 420–422, 2006. DOI: 10.1109/LCOMM.2006.1638602.

[100] Mathworks, "OFDM modulation," [Online]. Available: https://www.mathworks.com/help/lte/ref/lteofdmmodulate.html.

[101] S. Haene, A. Burg, N. Felber, and W. Fichtner, "OFDM channel estimation algorithm and ASIC implementation," in *2008 4th European Conference on Circuits and Systems for Communications*, 2008, pp. 270–275. DOI: 10.1109/ECCSC.2008.4611691.

[102] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, 2017. arXiv: 1412.6980 [cs.LG].

[103] M. Costa, "Writing on dirty paper," *IEEE Transactions on Information Theory*, vol. 29, no. 3, pp. 439–441, 1983. DOI: 10.1109/TIT.1983.1056659.