Domain aware deep learning for wireless physical layer

Shuvam Chakraborty
University at Albany, State University of New York, schakraborty@albany.edu

The University at Albany community has made this article openly available. Please share how this access benefits you.

Follow this and additional works at: https://scholarsarchive.library.albany.edu/legacy-etd

Part of the Computer Engineering Commons, and the Electrical and Electronics Commons

Recommended Citation
https://scholarsarchive.library.albany.edu/legacy-etd/2875

This Master's Thesis is brought to you for free and open access by the The Graduate School at Scholars Archive. It has been accepted for inclusion in Legacy Theses & Dissertations (2009 - 2024) by an authorized administrator of Scholars Archive.
Please see Terms of Use. For more information, please contact scholarsarchive@albany.edu.
DOMAIN AWARE DEEP LEARNING
FOR WIRELESS PHYSICAL LAYER

by

Shuvam Chakraborty

A Thesis
Submitted to the University at Albany, State University of New York
in Partial Fulfillment of
the Requirements for the Degree of
Masters of Science

College of Engineering and Applied Sciences
Department of Electrical and Computer Engineering

Fall 2022
ABSTRACT

Wireless receiver design for OFDM systems is well investigated with classical signal processing tools, which lack the capacity to extract intrinsic channel effects in received signal and lead to high decoding error in receiver. Current deep learning techniques have shown improvement in such cases. But these models are mostly being developed as black box without any anchor to the theory of wireless signal propagation, which leads to surface level information gain and lacks generalizability. We propose deep learning models where the hyperparameters and learning objectives are derived from domain knowledge of wireless signal propagation. These models not only increase the quality of channel estimation and equalization due to their capability to learn precise nonlinear functions, but take care of the primary drawbacks of popular deep learning models such as impractical data need and frequent retraining as well. On the other hand, for emerging spectra such as Terahertz band, that are in investigative stages currently, can benefit from such models as well. Since, the models are developed exploiting the fundamental features of wireless signal propagation, they help explore the channel and signal features of Terahertz band to develop an efficient and practical channel estimation technique and subsequently an end to end physical layer receiver design.
To My Parents
ACKNOWLEDGMENT

I would like to acknowledge some very special people who are the integral parts of my journey this far. I would like to begin by thanking my advisor, Prof. Dola Saha for the encouragement, advice and guidance she provided through my time as a graduate student. Our conversations in countless in-person and virtual meetings have always helped me to find a clear path out of any stagnation. You have helped shape my understanding and approach to a research project. Thank you for believing in me from the first day I joined the graduate school at University at Albany right after my undergrad and bearing with me during the hard times of the global pandemic. I am also grateful for the insightful comments and guidance provided by Prof. Aveek Dutta through the scope of our collaboration, which have helped enrich my research. I would like to thank Dr. Ngwe Thawdar and Claire Parisi from Air Force Research Lab, Rome NY for their support and cooperation during the tenure of our collaboration. I would also like to thank Prof. Hany Elgala and Prof. Aveek Dutta, for serving as my committee members.

I would also like to thank my colleagues and friends here in UAlbany and back home for their help, support and motivation during this time.

Finally, yet most importantly, Thank you Ma (Mom) and Papa (Dad) for your unwavering support throughout the journey until this day and days to come. I am here only because of you. Any word of thanks or gratitude will fall short for you two being my support system during my highs and lows, and trusting me with my decisions and accepting my successes and failures whole-heartedly.
# CONTENTS

**ABSTRACT** .......................................................... ii
**ACKNOWLEDGMENT** ........................................ iv
**LIST OF FIGURES** ............................................... ix
**LIST OF TABLES** ............................................... xi

1. Introduction .................................................. 1
   1.1 Sub-6 GHz Spectrum ....................................... 1
   1.2 Exploring Emerging Spectra .............................. 2
      1.2.1 THz Waveform Design ............................... 3
      1.2.2 Receiver design for THz band ................... 4
   1.3 Contribution .............................................. 5
   1.4 Previously Published Content ......................... 6
   1.5 Thesis Organization ..................................... 7

2. Background ................................................. 8
   2.1 OFDM .................................................... 8
      2.1.1 Transmitter ....................................... 8
      2.1.2 Receiver ......................................... 9
   2.2 Neural Networks ......................................... 13
      2.2.1 Fundamentals ..................................... 13
      2.2.2 Activation Function ............................ 14
      2.2.3 Architectures .................................... 15
      2.2.4 Complex Neural Networks ....................... 18

3. Related Work ............................................... 20
5.1 Contribution .......................................................... 39

5.2 A Case for OFDM .................................................. 40

5.2.1 Effect of Transceiver Hardware ......................... 42

5.2.2 Coherence Bandwidth ..................................... 42

5.2.3 OFDM as a Candidate Waveform ................. 43

5.2.3.1 Subcarrier Width ........................................... 44

5.2.3.2 FFT length ................................................... 44

5.2.3.3 Cyclic Prefix ................................................. 44

5.3 Derivation of OFDM Parameters ..................... 45

5.4 Implementation Details ..................................... 46

5.4.1 Baseband OFDM: ............................................. 46

5.4.2 Testbed Overview .......................................... 47

5.4.3 Testbed Specific Modifications: ................. 48

5.5 Experimental Results ......................................... 48

5.6 Conclusion ........................................................ 51

6. Communication Knowledge Aided Neural Network for OFDM Receiver in Terahertz Band ........................................... 52

6.1 Contribution ........................................................ 52

6.2 Physical Layer Receiver Design ....................... 53

6.2.1 Refined Channel Estimation (RCE) Block .......... 54

6.2.2 Enhanced Equalization (EE) Block ................. 56

6.3 OFDM Receiver in THz - A Case Study .............. 58

6.4 Experimental Setup ............................................ 59

6.4.1 Baseband Processing ....................................... 59

6.4.2 THz Testbed ................................................... 60

6.4.3 Neural Network Implementation ................. 60

6.5 Complexity Analysis ........................................... 61
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>THz spectrum compared to current radio spectrum</td>
<td>3</td>
</tr>
<tr>
<td>2.1</td>
<td>Basic OFDM Transceiver Blocks</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Subcarriers in OFDM symbol. Pilots are inserted at subcarrier indices -21, -7, +7, +21. The DC subcarrier at index 0 is left empty.</td>
<td>9</td>
</tr>
<tr>
<td>2.3</td>
<td>QAM 16 modulated symbols in presence of frequency selective fading.</td>
<td>12</td>
</tr>
<tr>
<td>2.4</td>
<td>What is Deep learning and Neural Networks?</td>
<td>15</td>
</tr>
<tr>
<td>2.5</td>
<td>Non-linear Activation Functions</td>
<td>16</td>
</tr>
<tr>
<td>2.6</td>
<td>Visual representation of different NN architectures. Starting from top: 1) Fully Connected Neural Network (FCNN), 2) Variation Auto Encoder (VAE), 3) Convolutional Neural Network (CNN).</td>
<td>17</td>
</tr>
<tr>
<td>2.7</td>
<td>Convolution operation in one layer with Max-Pooling.</td>
<td>18</td>
</tr>
<tr>
<td>2.8</td>
<td>Simplified block diagram of a single layer of complex network and its equivalent real network.</td>
<td>19</td>
</tr>
<tr>
<td>4.1</td>
<td>Domain Knowledge as input to the Neural Network Architecture for Channel Estimation.</td>
<td>25</td>
</tr>
<tr>
<td>4.2</td>
<td>Comparing how noise sensitivity decreases in the sample $Z$ from latent space $L$.</td>
<td>35</td>
</tr>
<tr>
<td>4.3</td>
<td>Performance of Denoising Block against generic feed forward network.</td>
<td>35</td>
</tr>
<tr>
<td>4.4</td>
<td>Performance comparison of denoising block with respect to fully connected network with training data belonging to all SNR ranges and selective SNR range.</td>
<td>37</td>
</tr>
<tr>
<td>4.5</td>
<td>MSE comparison of proposed domain knowledge based model with legacy methods.</td>
<td>37</td>
</tr>
<tr>
<td>5.1</td>
<td>Attenuation in Electromagnetic spectra in THz band.</td>
<td>40</td>
</tr>
<tr>
<td>5.2</td>
<td>The difference of attenuation over 10 GHz band at different transmit frequencies and distances.</td>
<td>41</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.3</td>
<td>Power Spectral Density of single carrier waveform in a bandwidth of 10GHz.</td>
<td>41</td>
</tr>
<tr>
<td>5.4</td>
<td>Frequency correlation of THz channel.</td>
<td>45</td>
</tr>
<tr>
<td>5.5</td>
<td>Block diagram presenting different components of the Implementation</td>
<td>46</td>
</tr>
<tr>
<td>5.6</td>
<td>The testbed equipment setup.</td>
<td>47</td>
</tr>
<tr>
<td>5.7</td>
<td>Error Vector Magnitude over Transmitter (AWG) amplitude for different guard length.</td>
<td>49</td>
</tr>
<tr>
<td>5.8</td>
<td>Bit Error Rate (BER) variation over received SNR.</td>
<td>50</td>
</tr>
<tr>
<td>6.1</td>
<td>Multi-stage Neural Network for OFDM receiver.</td>
<td>53</td>
</tr>
<tr>
<td>6.2</td>
<td>A case study for training data requirement.</td>
<td>55</td>
</tr>
<tr>
<td>6.3</td>
<td>Mechanism of EE block data generation.</td>
<td>56</td>
</tr>
<tr>
<td>6.4</td>
<td>Block diagram presenting different components of the Implementation</td>
<td>59</td>
</tr>
<tr>
<td>6.5</td>
<td>Evaluation of hypothesis and comparison with other possible receiver designs.</td>
<td>63</td>
</tr>
<tr>
<td>6.6</td>
<td>BER of NN model compared to DSP based model for different modulations</td>
<td>65</td>
</tr>
</tbody>
</table>
LIST OF TABLES

4.1 Different channel models used for experiment. . . . . . . . . . . . . . . . . . . 32
4.2 Training data used in different networks. . . . . . . . . . . . . . . . . . . . . . 35
5.1 Physical layer data rate at different modulation and $N_{CP}$. . . . . . . . . . 50
CHAPTER 1

Introduction

As the fractured sub-6 GHz spectrum gets congested \[\text{[1]}\] for wireless communication, and emerging applications, like Virtual Reality, telesurgery, etc. demand high bandwidth to support very high datarate of these applications, we need to rethink the physical layer models to meet such astronomical increase in data rate requirement. Current receiver design techniques in practice add an upper bound in datarate due to limitations in accurate channel estimation and equalization techniques. To overcome this bottleneck and achieve the desired datarate, two possible directions towards this goal are: 1) Update the receiver design in practice for sub-6 GHz spectrum to achieve higher data rate with reliability even in hostile channel conditions, 2) Venture out to other regions of the electromagnetic spectrum and explore emerging spectra. These possible solutions come with their own baggage but with a greater reward in terms of datarate performances. Possibilities, limitations, and viable solutions for either direction are discussed in this thesis.

1.1 Sub-6 GHz Spectrum

In the current RF transceiver systems, the methods used for multistage operation of time domain passband signal generation and decoding to obtain data bits are well-defined and benchmarked. But, the several channel estimation techniques \[\text{[44]}\] conceived have limited capability to extract channel information and efficiently use it for decoding in receivers, which leaves room for improvement. To name a few, the Least Square (LS) estimate is mostly implemented in Orthogonal Frequency Division Multiplexing (OFDM) signals for its low complexity. Minimum Mean Squared Error (MMSE), although provides high accuracy in estimation, is computationally complex and requires prior knowledge of the channel, which makes it impractical. It is evident that we require non-linear estimation techniques, which can perform beyond linear estimators like LS and aid the equalization process to improve decoding accuracy. Such non-linear optimal estimators are hard to find that is adaptable to constantly varying wireless channel. This compels us to innovate in novel accurate channel
estimation techniques that promise to yield higher data rate from a wireless channel.

Recent years have experienced tremendous growth in Deep Learning research and its capabilities are leading to breakthroughs in several domains of science and technology. Data driven deep neural networks (DNNs) are known to learn complex non-linear functions and may require less computation to approximate certain functions. Pure data driven models rely on the huge amount of data to train the models. On the contrary, introduction of domain knowledge has been shown to speed up the learning process of the vanilla data driven models and reduce the uncertainty in the learning model. This domain knowledge can be applied in different forms, like the choice of hyper-parameters, modification of the loss function, or adjustment of the weights in the layers. Although domain knowledge has been used in other scientific areas, and several data driven approaches have been explored for wireless communication, very limited work has been done that uses expert knowledge in neural network architecture or optimization for wireless communication that would minimize extensive training required for model accuracy. In this thesis, we introduce novel domain knowledge aided neural network (NN) architectures bridging the gap between model based and data driven approaches, such that we can exploit the benefits of both techniques for a system that is tractable and implementable in practical scenarios as further discussed in chapter 3. Instead of a black box approach of trying different NN models and retraining continuously for different practical scenarios, we start our model design with understanding of the source of error in wireless channels. Our NN design is based on the possible boundary values of channel parameters. Through a limited training process, the proposed models learn the statistical properties of the wireless signal and channel and not instantaneous variations, thus only needing limited training for different propagation scenarios.

1.2 Exploring Emerging Spectra

Although there are possibilities to increase data rate in the sub-6 GHz spectrum that is theoretically limited by Shanon’s channel capacity, which cannot be overcome. This leads us to explore other available spectra at a higher frequency. As correctly identified in [21], THz band, the last piece of the RF spectrum puzzle, is highly promising as it offers large chunks of contiguous unused bandwidth as shown in Figure 1.1. It is envisioned as one of the leading technology to meet the exponentially growing datarate requirements of
emerging and future wireless systems. To embrace the opportunities of wider swaths of this new spectrum, we first have to tackle the challenges posed by THz frequencies and then utilize its full potential for wireless communications. Unlike the sub-6 GHz spectrum, waveform design and hardware development for the THz band are not benchmarked. Hence, we start by looking at developing a waveform ideal to tackle the hostilities offered by the wireless environment in the THz spectrum and then further evolve the receiver design as needed.

1.2.1 THz Waveform Design

Attenuation in THz frequencies is predominantly caused due to high path loss and atmospheric absorption by oxygen and water vapor. It dictates the viability of indoor/low-distance communication in the THz band. Although small scale fading such as scattering and multipath may exist, their effects are nominal compared to the large scale fading. This is even more evident when the links are highly directional and transmitter and receiver antennas are aligned in a line-of-sight (LoS) path. Practical indoor channel measurements in THz frequencies also affirm that the path loss component dominates the channel model in LoS short range directional links in indoor environments. Although several waveforms have been designed theoretically for ultra-broadband THz frequencies, there is limited or no design of waveforms for practical hardware setup. In our effort
for waveform design, we have used THz transceivers at the Air force Research Lab (AFRL) facility with 10 GHz bandwidth and performed channel measurements as a first step. We noticed significant frequency selective fading, which is much higher than the theoretical value of cumulative attenuation caused by pathloss and absorption loss. We relate this selective fading to the several components of the hardware that introduces non-linearity at ultra-broad bands. Orthogonal frequency division multiplexing (OFDM) has been developed to address frequency selective fading caused due to multipath in sub-6GHz bands. It is also utilized in THz frequencies [7, 25] to address frequency selectivity due to absorption loss in longer distances. Additionally, OFDM promotes higher datarate by using closely packed orthogonal bands, which aligns with our final goal. We take this opportunity to derive the coherence bandwidth of the system and design OFDM parameters for hardware induced frequency selectivity in the received signal. This way we experimentally generate sufficient conditions for the OFDM parameters, develop the OFDM transceiver system in THz band, and verify with over-the-air transmission using the THz testbed.

1.2.2 Receiver design for THz band

The classical wireless communication models have been designed for sub-6 GHz frequencies, with limited bandwidth and minimal absorption loss compared to THz frequencies. Also, there is negligible variation in any losses, if at all, within a given bandwidth, and performance variation is dictated by multipath and scattering in the wireless environment. On the contrary, THz bands are much wider, even wider than millimeter-wave bands, capable of reaching up to hundreds of GHz. Naturally, this unforeseen wide bandwidth has its own limitations. Non-linearities get introduced in the system due to hardware impairments that can not be mitigated by known digital signal processing (DSP) tools. Channel estimation techniques like linear minimum mean squared error (LMMSE) [64] estimate and maximum likelihood estimate provide improved performance. However, their high computation cost and dependence on prior channel knowledge render them infeasible in practice. Some lower complexity approximations [20, 28, 43, 58] have been developed. However, they fail to improve beyond a limit as several assumptions, like channel follows a known distribution (e.g. Rayleigh), limited error in channel covariance and SNR estimation, do not hold for non-linearities introduced by hardware impairments in ultra-broadband channel. Furthermore, these receiver designs do not consider the correlation between in-phase (I) and quadrature-
phase (Q) signals, which gets introduced in the transceiver hardware.

We introduced OFDM waveform \cite{9} to tackle several issues like dominant propagation and frequency selectivity from of THz band, which falls short to provide desired high decoding accuracy due to the limitations in the traditional receiver mentioned above. Breakthroughs in several domains of wireless communication \cite{47} using neural networks due to their capability to substantially improve over the traditional communication modules inspired us to look into this aspect of deploying neural networks for our goal. Recent work \cite{70} has also developed a comprehensive neural network (NN) based wireless receiver where standard receiver sub-blocks are individually replaced by a neural network. However, most of these are tested with simulated wireless channel parameters, which lacks to show the effects of ultra-broadband channels on over-the-air captured signals. Although models in sub-6 GHz are well developed with decades of research, models for the combined effect of channel and hardware, do not exist for ultra-broad THz bands. This urges us to invest in developing a neural network model for an OFDM receiver that can account for this combined effect by inspecting over-the-air signals. Moreover, neural networks have originated mainly from image processing domain, where the underlying principles of the models are significantly different from those required for wireless communication. This is due to the difference in the type of data: red, green, and blue real components in images compared to real and imaginary components of complex domain wireless signals. Challenges also stem from the colossal amount of data required to train a NN that is blindly designed to tackle data generated from combinations of all possible modulation orders for large FFT sizes at different channel conditions. These two major challenges necessitate the injection of domain knowledge for wireless communication in our proposed NN model as it has the potential to learn intrinsic statistical properties from limited data \cite{32}.

1.3 Contribution

Contributions of the work presented in this thesis can be summarised as the following:

1. Design a channel estimation model for wireless systems leveraging the power of domain knowledge and data driven approach that can improve the receiver accuracy beyond the current methods and achieve a higher data rate while also being adaptable to
different propagation and system conditions.

2. Develop a multicarrier OFDM waveform suited for THz band communication using practical channel measurements and validate the same using an operating testbed.

3. Analyze unique channel conditions and issues with THz channel and develop a neural network model empowered by the theory of wireless communication to overcome the bottleneck while being adaptable to different system parameters and propagation conditions.

1.4 Previously Published Content

The following publications were a direct result of the work presented in this thesis and collectively constitute the complete domain aware deep learning for wireless receiver design research primarily conducted by me.


1The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant: 1) Include the following copyright notice in the references:[© year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication], 2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis online, 3) When posting your thesis on your university website, include the following message: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [name of university or educational entity]’s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

2Authors can include partial or complete papers of their own (and no fee is expected) in a dissertation as long as citations and DOI pointers to the Versions of Record in the ACM Digital Library are included.
1.5 Thesis Organization

The thesis is organized in the following way. Chapter 2 provides a theoretical background of OFDM, fundamentals of deep learning and neural network, which define the primary components and theme of this thesis. In chapter 3, the state of the art and other relevant publications are presented briefly. Chapter 4 elaborates on the domain knowledge aided channel estimation technique for sub-6 GHz spectrum. In chapter 5 we present the development of an experimental THz band waveform followed by receiver design for the THz band presented in chapter 6. Chapter 7 concludes the thesis.
CHAPTER 2

Background

In this section, we will introduce fundamentals and some background information related to the tools and methodologies used in the work presented in this thesis.

2.1 OFDM

Multi-carrier modulation has been the most prevalent form of communication in the recent decade. Orthogonal frequency division multiplexing (OFDM) is a specific implementation of multi-carrier modulation, that has been the default mode for the majority of communication systems across the world since the inception of 3rd Generation Partnership Project (3GPP) due to high datarate and robustness to inter-symbol and inter-carrier interference caused by frequency selective fading.

Figure 2.1 represents the transmitter-receiver blocks of an OFDM system. In this work, the signal structure of a packet-based WLAN system is adopted for system design and evaluation. So, further details of OFDM will include additional specifications from that. We discuss the sub-blocks and their inter-relations briefly here.

2.1.1 Transmitter

In OFDM, the input data bits are modulated and mapped into densely placed parallel subcarriers where subcarrier width and spacing are maintained in a way that the subcarriers are orthogonal to each other leading to a high data rate. The entire bandwidth is distributed

![Figure 2.1: Basic OFDM Transceiver Blocks.](image-url)
Figure 2.2: Subcarriers in OFDM symbol. Pilots are inserted at subcarrier indices -21, -7, +7, +21. The DC subcarrier at index 0 is left empty.

among 64 subcarriers indexed as -32 to +31 including the empty DC subcarrier. Out of these 52 are non-empty subcarriers including 48 data subcarriers and 4 pilot subcarriers located at specific indices: -21, -7, +7, and +21. Pilots are important to aid in channel estimate and phase correction at the receiver. After subcarrier mapping, IFFT is performed to get the time domain symbols. The cyclic prefix is appended to the time domain symbols before converting from parallel to serial. This is the complex modulated time domain baseband signal which is converted to a real passband signal before transmission at the frequency $f_0$. These signals are generally transmitted in a packet form that contains an integral number of OFDM symbols along with other requirements such as training symbols (preambles), and signal symbols generally preceding the data symbols in a packet. We can represent the general model of an OFDM signal at a carrier frequency $f_0$ as:

$$x_R(t) = \text{Re} \left\{ \sum_{k=-N_{\text{FFT}}/2}^{N_{\text{FFT}}/2} \alpha_k e^{j2\pi k(t-t_0)/T_u} \right\}$$ (2.1)

Where $\alpha_k$ is the m-ary modulated symbol (m depends on modulation order) which is placed on $k_{th}$ subcarrier, $T_u$ is the symbol duration and $k$ is the number of subcarriers. $x_R(t)$ represents the passband transmit signal.

### 2.1.2 Receiver

1. **Packet Detection:**

At the first step in the receiver, synchronization is of absolute importance to detect the symbols correctly and eventually decode the received bits. For that, two synchronization
method is adopted: 1) correlation with known symbols (e.g., preambles), 2) correlation with previously received symbols. In this work, we adopted correlation with preambles to detect the start of packets. Here, the receiver performs cross-correlation with the preamble. Correlation determines the degree of similarity in signals ranging from 0 (no similarity) to 1 (identical). This property is utilized to detect the start of the packet. Additionally, the data symbols are preceded by known preambles each of which occupies the same symbol duration as data symbols. Thus the start of data symbols can be easily detected once the start of the packet is detected using the cross-correlation method.

2. Channel Estimation and Equalization:

Channel fading is a common phenomenon in wireless communication, which distorts the signal in a non-linear manner, i.e., both amplitude and phase get distorted. Channel fading can be flat - channel fading is similar across the bandwidth, frequency selective - across bandwidth different frequencies have different levels of fading. To mitigate this, preambles and/or pilots are used to estimate the channel. Since preambles are spread across the entire bandwidth, they can provide a rigorous channel estimate using the known transmitted preambles. If preambles are not present (they are specific to WLAN), carefully placed pilot symbols shown in Figure 2.2 are used for channel estimation. channel state information is extracted from the pilots and using linear or non-linear interpolation methods final channel state is estimated. The most used methods of channel estimation in OFDM are presented here. The $i^{th}$ received symbol can be denoted as,

$$y_i = h_i \circ x_i + z_i, \quad y_i \in \mathbb{C}^{K \times 1}$$  \hspace{1cm} (2.2)

where $h_i$ is the channel impulse response, $x_i$ represents the $i^{th}$ transmitted symbol, $\circ$ denotes Hadamard product operation and $z_i \sim \mathcal{N}(0, \sigma^2)$ denotes the additive noise. Performing DFT, we achieve the frequency domain representation of the received symbol as,

$$Y_i = \tilde{H}_i X_i + Z_i$$  \hspace{1cm} (2.3)

where $\tilde{H}$ is the channel frequency response and $X_i$ is the transmitted $i^{th}$ data symbol. Since we are dealing with the WLAN packets, it contains the training symbols or preambles (short and long) that occupy two symbol slots each at the beginning of a transmitted data packet.
and are used for channel estimation purposes. So, we can express the received preamble as:

\[ Y_p = \tilde{H}_p X_p + Z_p \]  

(2.4)

where \( X_p \) and \( Y_p \) are the transmitted and received preambles, \( H_p \) is the channel response encountered by the signal and \( Z_p \) is additive noise. The least square estimate (LS) of the channel is obtained by optimizing the squared error which is given in equation equation (2.5):

\[ \tilde{H}_{LS} = X_p^{-1}Y_p \] 

(2.5)

LS estimation is the simplest method of channel estimation widely used in wireless systems due to its low computation complexity at the cost of low accuracy. Minimum mean square error (MMSE) estimation in equation equation (2.6) on the other hand provides the best possible accurate estimate since it utilizes prior channel information in the form of the channel covariance matrix. But this method presents two primary challenges: 1) prior channel knowledge is not available in most practical scenarios, and 2) it has very high computation complexity due to the need to compute matrix inverse and several matrix multiplications.

\[ \tilde{H}_{MMSE} = R_{H_p}(R_{H_p} + (X_p X_p^H)^{-1}\sigma^2 I_P)^{-1}.\tilde{H}_{LS} \] 

(2.6)

\( R_{H_p} \) is the channel covariance matrix which depends on prior channel knowledge, \( I_P \) is an identity matrix of size \((P \times P)\) and \( \sigma^2 \) is the noise variance. In a broad sense we can express the MMSE estimate as an error correction on top of the least square estimate as is evident from its expression.

These estimates are used to equalize the data subcarriers in order to reinstate their modulation levels to recover the data bits. Equalization in OFDM is generally performed as one tap LS equalization as presented inequation (2.7):

\[ \tilde{X}_{LS} = H_{LS/MMSE}^{-1}Y_i \] 

(2.7)

3. Practical MMSE:

For evaluating purposes in later chapters, we introduced LS and MMSE estimates as
benchmarks. To provide additional comparison we have also introduced the practical MMSE method proposed in [20]. It is an approximation of the MMSE estimator that compromises some accuracy in the estimate to reduce high complexity of MMSE estimation. MMSE estimate is expressed as:

\[
\tilde{H}_{MMSE} = R_{H_p} (R_{H_p} + (X_pX_p^H)^{-1}\sigma^2I_P)^{-1}.\tilde{H}_{LS}
\] (2.8)

This method has a very high complexity due to several matrix multiplications and matrix inversion. Given the pilots are predefined for each symbol, an approximation can be done by replacing \(X_pX_p^H)^{-1}\) with \(E(X_pX_p^H)^{-1}\), where \(E(X_pX_p^H)^{-1} = E(1/|x_p|^2)\). To approximate, this channel covariance is decomposed based on singular value decomposition (SVD) as:

\[
R_{H_p} = U\Lambda U^H
\] (2.9)

Where \(U\) contains the orthonormal eigenvectors and \(\Lambda\) is a diagonal matrix containing the corresponding eigenvalues \((\lambda_1 \geq \lambda_2 \geq \ldots \geq 0)\). From this, equation (2.8) boils down to

\[
\tilde{H}_{MMSE} = U\Delta U^H .\tilde{H}_{LS}.
\]

\(\Delta\) contains another set of eigenvalues \(\delta_k = \lambda_k/(\lambda_k + \beta/SNR)\), where \(\beta = E|x_k|^2e|1/x_k|^2\). Choosing a set of top d eigenvalues from \(\Delta\), the approximate estimate becomes:
\[
\tilde{H}_{\text{PracticalMMSE}} = U \begin{bmatrix}
\Delta_d & 0 \\
0 & 0
\end{bmatrix} U^H \bar{H}_{\text{LS}}
\] 

(2.10)

where \(\Delta_d\) contains top \(d\) eigenvalues. This way, not only the matrix inversion is avoided, but the overall matrix size for matrix multiplication is also reduced at the cost of some estimated channel energy lost in the form of discarded lower eigenvalues.

4. **Demodulation**: The demodulation is performed after the equalization. The equalizer recovers the channel-distorted constellation and restores it to close to its original configuration. The demodulation is achieved by a maximum likelihood estimator in baseband, which performs a threshold test as per the symbol energy. Decision boundaries are given by the perpendicular bisector of the line joining the two constellation points. This gives the optimum decoder as it minimizes the Euclidean distance between the received signal and the nearest constellation point and decides the coded bits based on that. Two forms of decoding is generally adopted in OFDM: hard-decision decoding and soft-decision decoding.

### 2.2 Neural Networks

#### 2.2.1 Fundamentals

The underlying concept of deep learning [40] is using deep neural networks (multiple hidden layers) to learn any hidden pattern from the data and approximate highly non-linear functions with great accuracy. Figure 2.4a presents one of the most well established contextual definition of deep learning.

In Figure 2.4b, the inner working of a single neuron is presented which is the fundamental unit of any neural network. Input to a Neuron is a tensor which is presented here in the figure as a 1-dimensional tensor \(X\). It is multiplied by the randomized weight \(W\) and bias \(B\) is added. Bias plays an important role here. It provides additional flexibility to the neuron output. The output is then passed through an activation function. As the name suggests, activation functions decide the forward propagation of the neuron output. Details of activation functions are discussed later. Up to this encompasses the forward pass of data through a single neuron. The fundamental goal of using NN is to reach as close to its objective as possible using optimal network weights and biases. The objective is generally expressed as
an objective function or loss function. It denotes the error or distance from the objective in some metric. For optimizing a neural network, the backpropagation algorithm [57] is used in an iterative manner. Using the example of a single neuron we present the mathematical formulation. Assuming a generic loss function $\mathcal{L}(Y, \hat{P})$ where $\hat{P}$ can be any parameter as a part of the definition of the loss function (e.g., true labels for supervised learning), we get updated weight for $(n+1)^{th}$ step as:

$$W_{n+1} = W_n - \eta \frac{\delta \mathcal{L}(Y, \hat{P})}{\delta W}$$

(2.11)

Where $W^n$ is the weight at the $n^{th}$ step, $\eta$ is the learning rate which controls how fast the gradient is updated and the gradient of the loss function with respect to the weight provides the direction towards which the weight should change, assuming the loss as the function of the network weights. This is performed for each neuron in a neural network in an iterative manner to achieve the optima. Backpropagation is performed with the assumption that the objective and the dataset present a convex optimization problem and if this is violated, the network can get stuck at a saddle point. In a deep neural network, often optimizing each data point at each neuron becomes computationally sub-optimal. To avoid that optimization is performed in batches - a subset of the whole dataset randomly sampled at each iteration. For gradient upgrade, over the years several updated optimization techniques have been adopted. Out of which stochastic gradient descent (SGD) [34] and ADAM [36] optimizer have been proven most effective and heavily used in deep NN applications. As we observe the neural network architecture in its basic form, it is important to understand, that in a deep neural network there are many layers and even more neurons. The hyperparameters in those cases like number of layers and neurons, learning rate, and batch size need to be very carefully chosen to achieve the desired objective.

2.2.2 Activation Function

Activation functions govern the passage of outputs of the neurons from each layer of a deep neural network. They are primarily classified as linear and non-linear activation functions. Using Linear activation functions simply indicates that there are no additional assumptions and conditions on the layer output. Non-linear activations however are of utmost importance here. They not only control the forward passage in NN, they are the
source of non-linearity in a neural network that helps the networks approximate non-linear functions of high degree. Choosing the right activation function for each layer keeps the back-propagation in check by preventing vanishing or exploding gradient problems. We have graphically presented four different non-linear functions: Sigmoid, tanh, Rectified Linear Unit (ReLU) and Leaky ReLU in Figure 2.5 along with their mathematical form. Use of these functions in many cases is subjective to the application or objective of the network. For example, sigmoid function is more suited for classification tasks as it is capable of providing the probability estimate of a class for a given input. On the other hand, Tanh is better suited for regression objectives.

2.2.3 Architectures

Over the decade of development, NN architectures have taken several forms like convolutional networks (CNN) for spatial correlation, recurrent NNs (RNN) that find temporal correlation and learn from present and past states of the objective, auto-encoders and generative adversarial networks (GAN) for generative tasks, etc. Here we present the fundamentals of the most relevant NN architectures used in the development of this thesis. A comprehensive overview of these networks along with the graphic comparison of these network architectures are presented in Figure 2.6.

1. **Fully Connected Neural Network**

A fully connected neural network (FCNN) is an extended version of the single neuron structure explained before with multiple hidden layers and multiple neurons in each layer.
(a) Sigmoid: \( f(x) = \frac{1}{1+e^x} \).

(b) Tanh: \( f(x) = \frac{e^x-e^{-x}}{e^x+e^{-x}} \).

(c) ReLU: \( f(x) = \max(0,x) \).

(d) Leaky RELU: \( f(x) = \max(ax,x), 0 < a < 1 \).

Figure 2.5: Non-linear Activation Functions.

It is primarily used for supervised learning (ground truth for the objective is known during training) tasks in an adversarial manner. FCNN weights are trained in the same way using the backpropagation algorithm as well. Being the most generalized version of NN architectures, FCNNs can be used for a multitude of tasks in both classification and regression problems. But, all neurons of a single layer being connected to all neurons of immediate neighboring layers makes it computationally inefficient.

2. Variational Auto-encoder

Variational auto-encoder (VAE) is a generative neural network consisting of two components: an encoder and a decoder. The aim of the encoder, an inference network, is to achieve an approximate posterior distribution \( q(L|Y) \), where \( Y \) is the input data and \( L \) is the latent space by minimizing Kullback-Leibler (KL) divergence between them. The encoder network produces mean \( \mu_L \) and standard deviation \( \sigma_L \) of the latent vector. The decoder, a
Generative network, aims to maximize the likelihood of \( p(\hat{Y} | Z) \), where \( Z \) is sampled from \( \mu_L \) and \( \sigma_L \) and \( \hat{Y} \) is the regenerated output. The two networks are parameterized with two different sets of parameters, \( \phi \) and \( \theta \). The loss function of VAE is given as:

\[
Loss(\theta, \phi) = \sum_{i=1}^{N} \mathbb{E}_{Z_i \sim q_{\phi}(Z | Y)}[\log(p_{\theta}(\hat{Y} | Z))] - KL(q_{\phi}(Z | Y)||p(Z))
\]

The first term represents reconstruction loss (decoder loss) of output \( \hat{Y} \) from \( Z \), which is the negative log-likelihood estimate. The second term captures the information loss due to the compression of data (encoder loss) into low dimensional latent space, where \( KL(||) \) represents KL divergence.

### 3. Convolutional Neural Network

A convolutional neural network (CNN) finds spatial and temporal correlation along
the dimensions of the input tensor by correlating kernels or filters with them. Generally, CNNs include multiple filters of specific shape, that are correlated along the pre-specified dimensions of the input data matrix by moving the filters for fixed units defined as strides along that dimensions. This provides an abstracted feature map of the data and extracts the correlation information. Based on the stride and kernel size, it is possible that the kernel is unable to correlate with certain parts of the input. To avoid that, appropriate padding of zeroes is used around the input data. This does not influence the feature map but allows the kernel to traverse across the input data dimensions more efficiently. Generally, we choose the filter size based on the expectation upto which we can find a significant correlation in the input tensor, which cannot be efficiently done using a fully connected network. Pooling layers are generally associated with CNNs as they help in reducing dimensionality of CNN outputs by either choosing the maximum (max pooling) or average (average pooling) across a pooling window. An example realization of these details and computations performed in a single layer of CNN is presented in Figure 2.7. Here the convolution layer is a 2-D layer as the kernel has both height and width > 1. In Figure 2.6 we have shown a full convolutional layer with 1-D kernels that perform in identical manner to 2-D kernels. 1-D kernels are of particular significance in this thesis due to their utilization to achieve different desired objectives have been described in later chapters.

2.2.4 Complex Neural Networks

In this section, we describe the inner workings of the complex-valued neural network used in chapter 6. A complex number \( z = a + ib \) has a real component \( a \) and an imaginary component \( b \). In complex neural networks, input, weights, and biases are all complex tensors.
and the activations are complex functions, although the forward and backward propagation are almost identical to their real counterparts. During computation, the real part and the imaginary part of the tensors are treated as distinct real entries and are plugged into the linear complex arithmetic. For example, if we assume a complex input $x_{m \times 1}$ and a complex weight in a layer $W_{m \times n} = W_R + iW_I$, we can represent the output $y_{n \times 1}$ of the layer in matrix form as:

$$
\begin{bmatrix}
Re(y) \\
Im(y)
\end{bmatrix} = \begin{bmatrix}
W_R & -W_I \\
W_I & W_R
\end{bmatrix} \begin{bmatrix}
Re(x) \\
Im(x)
\end{bmatrix}.
$$

(2.13)

$x, y \in \mathbb{C}$ are vectors and $W \in \mathbb{C}$ is a matrix. In comparison, a real network will require $\mathbb{R}_{2m \times 1}$ input and $\mathbb{R}_{2m \times 2n}$ to achieve the full expressive capability of the complex network. We have shown the simplified representation of a fully connected complex network and its equivalent real network with full expressive capability in a simplified block diagram without the activation in Figure 2.8. Also, this alternative real network introduces unnecessary degrees of freedom which can be avoided and increase efficiency using the complex network and still be able to explore the I-Q correlation. In addition to these features, for the back-propagation, it is necessary to guarantee that the non-linear activation functions and the objective for the complex neural network both are differentiable with respect to the complex weights for back-propagation. Fortunately, it is sufficient to guarantee differentiability of the activation and objective with respect to the real and imaginary part of the weights separately and they are not required to be complex differentiable or holomorphic as explained in [26]. This provides significant relaxation for the choice of activation and objective function and differentiability can be guaranteed easily with respect to real and imaginary components separately without going for further mathematical derivation and tuning the objective.
CHAPTER 3
Related Work

We have identified the existing literature relevant to the work presented in this thesis in two primary categories: Channel Estimation in Sub-6 GHz, THz communication system development, and additional sub-categories within them. We presented the state of the art for both waveform development and receiver design in THz under one umbrella due to their strong correlation and overlap in state of the art. We added one additional category as domain knowledge aided neural network separately as it has been a unique theme for the majority of the work presented here.

3.1 Channel Estimation in sub-6 GHz

Wireless channel estimation for sub-6 GHz bands has been well investigated leveraging traditional digital signal processing tools. Here we mention the most relevant ones in practice.

1) Model driven approach: Most model driven approaches exploit the training symbols (preambles and pilot tones) for time or frequency domain channel estimation. Authors in [64] have shown linear MMSE based channel estimation. To reduce the high complexity of MMSE approach, many approximation techniques, like Most Significant Tap (MST) [20] and Low Rank Approximation (LRA)-MMSE [43] have been proposed. More work [42] shows customized pilot based techniques to estimate fast varying channels. These models suffer due to near MMSE complexity or increased payload due to additional pilot tones.

2) Data driven approach: With the emergence of new machine learning and deep learning algorithms, there has been a swarm of applications using these tools in physical layer communication as well as channel estimation [47]. A fully connected DNN based approach has been proposed in [68] where channel estimate is captured from received data and pilots and signal detection is performed online. [49] shows a trade-off between accuracy and complexity of the MMSE estimator achieved by forcing the channel response matrix to achieve the Toeplitz structure using NN. Recent research [6] also borrowed neural networks developed for image classification [41] to perform channel estimation. [62] has used a convolution neural
network used for low resolution image detection, for channel estimation. Also, the authors in [15] have utilized sequential models as a black box approach. These transfer models may work well for a given scenario but do not guarantee performance accuracy in a wide variety of cases. The reasons for this limitation are a) image data and wireless signals do not follow identical distribution and b) knowledge of wireless signal propagation is not incorporated in model generation. [50] has represented DNN based physical layer model, with fading channel represented by a generative adversarial network (GAN). Authors have shown in [23], a shallow fully connected network can enhance LS channel estimate. It has been shown [27] that in an ideal setting with unrestricted training resources, DNN based channel estimators asymptotically reach the accuracy of MMSE estimator. However, we need to be careful in embracing these models as their timing and computation requirements are often impractical.

3.2 THz communication system development

1) Terahertz Communication for xG: As a frontier in the sub-terahertz and terahertz band communication literature, authors in [56] have discussed challenges and possibilities in communication beyond 100 GHz RF bands along with some state of the art results targeting different issues for NextG communication and beyond such as developing propagation and pathloss models beamforming requirement for long distance communication, reducing computation complexity, efficient beam steering and more. A set of complimentary results are presented in [13], focusing primarily towards the hardware requirement for terahertz band communication such as updated solid-state superheterodyne receivers, THz modulators, and antennas along with a few experimental systems developed recently. Besides physical layer development, novel changes are required for higher layers of the communication systems as well, which is discussed in [22] focusing on MAC layer protocol challenges, possibilities and developments for nextG communication. Use of intelligent reflecting surfaces (IRS), one of the most promising aid in THz band communication, are discussed in [14]. The authors proposed to leverage joint active and passive beam forming using IRS to tackle the severe propagation and absorption loss persistent in the THz band. A full stack end-to-end THz system development and the immediate link and system level issues on the way are discussed in [53]. As an eventual follow-up of the several state of the art developments for nextG communication is standardization efforts. IEEE 802.15.3d, first of its kind standardization for
subterahertz band discussed in [52] has paved the path for that.

2) THz Modelling and Waveform Development:

Terahertz (THz) frequencies offer large chunks of contiguous unlicensed spectrum for wireless communication that has the potential to support high datarate of emerging applications and meet the future spectrum need of xG communication. To model THz waveform understanding challenges posed by THz frequencies are of absolute importance [55]. Attenuation in THz frequencies is predominantly due to high path loss and atmospheric absorption by oxygen and water vapor [30]. Small scale fading such as scattering [31, 45] and multipath [46] effects are nominal compared to the path loss. Practical indoor channel measurements are performed in [3, 38, 69] that affirms dominance of large scale fading (primarily pathloss) in LoS short range directional links in indoor environments. Furthermore, system level experimental testbeds [59, 60] have been conceived for waveform design. Although several waveforms [25] have been designed theoretically for ultra-broadband THz frequencies, there is limited or no design of waveforms for practical hardware setup. Authors in [5, 18] have introduced predistortion to compensate for hardware impairments in radio signal. Availability of ultra-broad bandwidth in terahertz frequencies compels us to utilize it for wireless communication. Furthermore, our THz transceivers have much wider bandwidth (10 GHz) compared to prior testbed setups [3, 60]. Orthogonal frequency division multiplexing (OFDM) has been developed to address frequency selective fading caused due to multipath in sub-6GHz bands. It is also utilized in THz frequencies [7, 25] to address frequency selectivity due to absorption loss in longer distances.

3) Deep Learning in Wireless Receiver Design: There has been several applications of deep learning tools in wireless physical layer design [47]. Most channel estimation [6, 68] techniques using deep learning delve into one block of the receiver. Authors in [70] have developed a comprehensive neural network based OFDM wireless receiver, but in the equalization step, they treat each subcarrier as a separate input, which fails to explore the correlation of subcarriers within coherence bandwidth. [50] has represented DNN based physical layer model, with fading channel represented by a generative adversarial network (GAN). [33] suggests that the lack of reliable over the air training data for deep learning based wireless communication can be mitigated by synthetic data using a generative refiner model. This is subjective to the application and may harm the training for a nascent field
of study, like THz band communication, as it lacks benchmarks to rely on for synthetic data generation.

### 3.3 Domain Knowledge Aided Neural Network

Theory or domain knowledge aided data driven model is gaining attention due to its capability to create tractable and explainable models with limited data. Authors in [32] combined physics-based model with neural network architecture for lake temperature modeling. Knowledge of medicine from doctors has been extensively incorporated into deep learning models [67] for various medical imaging tasks. It is shown in [35] that leveraging the physics background allows near maximum likelihood (ML) accuracy in soft MIMO detection for 5G NR. Expert knowledge has also been introduced in material science [16] to improve the interpretability of the predictions.
CHAPTER 4

Domain Knowledge aided Neural Network for Wireless Channel Estimation

Channel estimation for OFDM in sub-6 GHz is well investigated with signal processing based models. Several recent works also have explored the data driven approaches to exploit the capabilities of NNs to estimate the channel. These models are mostly being developed as a black box without any anchor to the theory of wireless signal propagation leading to a sub-optimal performance in terms of accuracy or computation cost. We propose a NN model, where the structure and hyperparameters are derived from the domain knowledge of wireless signal and channel characteristics, bridging the gap between model based and data driven approaches. This way, we exploit the benefits of both classes of estimation techniques for developing a system that is tractable and implementable in practical scenarios. Instead of a black box approach of trying different NN models and retraining continuously for different practical scenarios, we start our model design by understanding the source of error in wireless channels. The proposed model is developed in two stages: the first stage handles the noise reduction, while the second stage extracts the channel information to reduce the error caused by frequency selective fading. Through limited training process, the proposed model learns the statistical properties of the channel instead of instantaneous variations. Channel estimation accuracy in terms of normalized mean squared error (NMSE) also shows that induction of domain knowledge results in reduction of data required in model training by 60%. This model outperforms the practical signal processing based methods as well as blind data driven approaches, achieving upto ~10 dB improvement over the Least Square channel estimate.

4.1 Contribution

The contributions of this work can be listed as:

1. Incorporation of domain knowledge: An NN architecture for channel estimation is presented, by systematically breaking the problem into two phases, based on the knowledge of
2. Signal to noise ratio (SNR) as input for denoising: The first stage of the channel estimation is a neural network, which denoises the initial channel estimates. We introduced a variational autoencoder model that modified the latent space with the SNR of the signal, such that reconstruction can be improved based on the posterior probability.

3. Coherence bandwidth to determine filter size: The second stage of the estimation technique resolves frequency selective fading caused by multipath effect using custom convolution layers. We formulate the effect of coherence bandwidth on the size of filters for convolution, which guides the architecture design of the NN. This formulation is generic and can be used for any communication channel.

4. Benefits of injecting domain knowledge: Our model outperforms any practical channel estimation techniques and requires fewer data and training iterations to reach better accuracy compared to black box models.

### 4.2 Two Phase Channel Estimator Model

In this section, we introduce the design and training approaches of the DNN based channel estimator for OFDM powered by domain knowledge. In this work, we broadly explored channel conditions in terrestrial communication where two predominant sources of distortions are: additive noise and inter-symbol (ISI) and inter-carrier interference (ICI) caused by frequency selective fading (multipath). So, we conceive a two-stage solution with specific tasks for each stage: A) Denoising Block, responsible to minimize the noise of the
initial estimate, followed by B) Multipath Resolution (MPR) Block, responsible to estimate and mitigate the channel effects causing ISI and ICI. The system takes the frequency domain LS estimate of channel as input and the output is the improved version of the estimate in frequency domain. Figure 4.1 shows the complete two-stage model, along with the domain knowledge utilized in their development.

4.2.1 Denoising Block

In a noisy signal, noise variance increases uncertainty in detection. This encouraged us to incorporate a probabilistic generative model, variational autoencoder (VAE) [37] due to its efficiency in generating a higher resolution noise-free version of a multi-dimensional noisy input. The system diagram of denoising block is shown in the green block of Figure 4.1.

4.2.1.1 Latent space manipulation of VAE with domain knowledge (SNR)

At receiver side, we can only measure a few parameters, SNR being one of those that dictate the channel characteristics. So, we will use the SNR information to update the latent space for our cause. It is shown in the yellow sub-block in Figure 4.1. LS estimate is the input to the VAE, which consists of both distortions: noise and interference due to multipath. Notations used here for mathematical expressions for the VAE are consistent with the ones used for introducing VAE in section 2.2.3. Let us Consider that additive white Gaussian noise (AWGN) ($\sim N(0, \sigma_n)$) is the primary noise distortion in the channel. As only white noise is present in data, it will be present in the compressed latent space as well. Since additive noise is zero mean, only variance of latent space $\sigma_L^2$ gets distorted by noise. So, for inputs at different SNR, $\sigma_L^2$ will have different levels of distortion, higher at low SNR and lower at high SNR. To, regenerate the channel estimate free from noise distortion, the sample space $Z$ needs to be generated from a distribution that does not display different levels of distortion varying with SNR. So, we use SNR to mitigate this issue. From definition:

$$\text{SNR} \propto 1/\sigma_n^2$$
We updated variance of the latent space by introducing SNR as:

\[
\sigma_{\text{denoise}}^2 = \text{SNR}\sigma_L^2 \implies \sigma_{\text{denoise}} = \sqrt{\text{SNR}}\sigma_L
\] (4.1)

Low SNR implies higher level of noise, which increases uncertainty in accurately estimating the posterior \(q_\phi(Z|Y)\), subsequently increasing \(\sigma_L\). Multiplying this term with \(\sqrt{\text{SNR}}\) is equivalent to dividing with the standard deviation of noise distribution, i.e., reducing sensitivity to noise. In this step, a measured SNR value at the receiver is used, due to which any effect of estimation error will have similar effect in both training and testing process. So, the model does not require any further calibration for estimation error in SNR. Moreover, as the domain knowledge based variance update reduces effect of noise, the network requires fewer data to train than a black-box approach. With the guidance of domain knowledge, the network can adapt itself without training at all practical SNR ranges. We will still need some data from different SNR ranges, as the posterior still contains approximation error, which can be reduced by training with data from different SNR ranges, as prescribed in section 4.3.2.

### 4.2.1.2 Updated Loss Function

The reconstruction loss in the loss function of VAE, presented in equation (2.12) is the negative log-likelihood:

\[
E_{(Z_i \sim q_\phi(Z|Y))} \log(p_\theta(\hat{Y}|Z)).
\]

The posterior, \(q_\phi(Z|Y)\), is modeled as Gaussian due to the primary assumption of VAE that evidence lower bound is approximated as Gaussian [37]. Since in this work we have expert channel estimates \(Y_{label}\) available, we can replace \(Z\) with these in the decoder loss. With a known distribution and ELBO approximating to Gaussian, the decoder loss boils down to mean squared error between \(\hat{Y}\) and \(Y_{label}\) as shown in equation (4.2):

\[
\text{Loss}_{\text{reconstruction}} = -E_{(q_\phi(Z|Y))} ||D_\theta(Z) - Y_{label}||^2
\] (4.2)

\[
\hat{Y}_{\text{denoise}} = D_\theta(Z), \quad Z \sim \mathcal{N}(\mu_L, \sigma_{\text{denoise}})
\]
where $D_\theta$ denotes the decoder network. The updated loss function of the denoising block is:

$$
\text{Loss}_{\text{denoise}}(\theta, \phi) = \sum_{i=1}^{N} -E_{q_\phi(Z|Y)}[||\hat{Y}_{\text{denoise}} - Y_{\text{label}}||^2]
- KL(q_\phi(Z|Y)||p(Z)) \tag{4.3}
$$

$$(\hat{\theta}, \hat{\phi}) = \arg\min_{\theta, \phi} (\text{Loss}_{\text{denoise}})$$

4.2.2 Multipath Resolution (MPR) Block

After denoising the signal, our subsequent goal is to extract subcarrier correlation information to mitigate the estimation error caused by multipath. The system diagram of MPR is shown in the blue block of Figure 4.1.

We chose convolutional neural networks for development of MPR block due to its efficiency in finding spatial correlation and feature map from the data. In our application, the input data is shaped as $N_f \times 2$ where $N_f$ is the number of subcarriers and the two channels contain I and Q samples. We needed to store them separately since complex tensor computation was not supported in our chosen implementation platforms at the time of experimentation. While implementing CNNs, we chose 1-dimensional convolution layers, as each data channel (channel of NN) had a shape of 1D tensor.

4.2.2.1 Coherence bandwidth and channel variation

Coherence bandwidth denotes the width of frequency band upto which channel fading remains flat. This idea of coherence bandwidth originates from channel delay spread. In a multipath fading channel, different paths incur different delay in reflected signal components. The root mean square (RMS) delay spread $t_{\text{rms}}$, dictates the extent of coherence bandwidth $B_c$ as $B_c \propto 1/t_{\text{rms}}$. So, in case of an OFDM transmission, we define another term $K_{sc}$ as:

$$
K_{sc} = \left\lceil \frac{B_c}{w_{sc}} \right\rceil \tag{4.4}
$$
where $w_{sc}$ is individual subcarrier width. Thus we can simply say that the channel varies beyond $K_{sc}$ adjacent subcarriers. This is the domain knowledge that we exploit to successfully extract multipath information from the channel frequency response shown in the yellow sub-block in Figure 4.1.

### 4.2.2.2 Resolving the multipath with local sparse network

When the channel changes beyond $K_{sc}$ subcarriers, it creates a sharp contrast in its features around the boundary region. The idea here is to use a 1D filter in the MPR block which can easily extract the channel information and find these regions of contrast similar to edge detection of a gray-scale image. Our goal is to efficiently choose a filter that does not increase computational complexity and efficiently extract the multipath information. Hence, we provide this boundary condition for filter length $w_f \in (1, K_{sc}]$, as an input parameter of the neural network. Filter length 1 is not effective as it is inefficient and only helps in dimensionality reduction (equivalent to only pooling).

### 4.2.2.3 Optimal filter size

For a practical channel, we cannot explicitly know $K_{sc}$ apriori, and thus we cannot update the best filter size in an online manner. So, we look at the terminal conditions to optimize it. This work is based on IEEE 802.11 based WLAN systems and thus the filter size criterion is chosen based on that. In general $K_{sc}$ and corresponding filter size need to be selected based on the communication protocol in use. In WLAN OFDM packets, each time-domain symbol is appended with a guard interval of duration $0.8\mu s$, that mitigates ISI caused by multipath effects, where $t_{rms} \leq 0.8\mu s$. In the worst case scenario, an urban outdoor channel with several multipaths, high delay spread, and strong attenuation, has an RMS delay spread $\sim 1.4\mu s$, which cannot be taken care of by the guard band. This indicates a coherence bandwidth of $\sim 0.9$ MHz, thus, $K_{sc} \approx 3$. So, from the previous boundary condition of filter length, we choose $w_f = 3$, which is optimal for most considerable channel conditions where WLAN is used. This minimizes the complexity of the network, as complexity of CNN $\propto$ filter size.

The MPR block generates the channel estimate $\hat{Y}_{MPR}$ from the input tensor which is the reshaped form of $\hat{Y}_{denoise}$, shown in Figure 4.1. MPR block is parameterized by $\psi$. The
The network representation is given as:

\[ \hat{Y}_{MPR} = MPR\psi(\hat{Y}_{denoise}) \]

The loss function of the MPR block is defined as mean squared error between reconstructed channel estimate \( \hat{Y}_{MPR} \) and true labels \( Y_{label} \).

\[ Loss_{MPR} = \sum ||Y_{label} - \hat{Y}_{MPR}||^2 \]  \hspace{1cm} (4.5)

\[ \hat{\psi} = \arg\min_{\psi}(Loss_{MPR}) \]

4.3 Experimental Setup

4.3.1 Network Architecture

We have specified input and output shapes for both networks in Figure 4.1. The number of neurons in each layer is \( kN_{sc} \), \( N_{sc} \) = number of active subcarriers, \( N_{sc} \leq N_f \), where \( k \in \mathbb{R}^{++} \) such that \( kN_{sc} \in \mathbb{N} \). In both networks, we have chosen Leaky Rectified linear unit (LReLU) as the non-linear activation function for the hidden layers except the output layers. ReLU activation is superior to other possible activations like \( tanh \) for two specific reasons: ReLU 1) successfully prevents gradient saturation (vanishing or exploding), and 2) is computationally less costly than other likely activation functions. But ReLU limits the output to positive values only, making it non-viable to use as I and Q samples of the input tensor includes negative numbers as well. So, we used leaky ReLU activations with 30% leak. The output layers of both denoising block and MPR has linear activation functions.

4.3.2 Training SNR

While dealing with channel estimation using neural networks, one of the crucial hyperparameters to choose is training SNR. Being completely experimental, there is no clear guidance for choosing that. Some of the existing works [70], suggested low to moderate training SNR, which is mostly experimental consideration. There are some benefits of training at low SNR. Presence of noise provides a natural regularizer minimizing overfitting, but it also suppresses other system errors as well, which may show up in high SNR validation results. So, we choose a range of SNR values to train the network. In case of a black box application,
to generalize the network well, we would need data from all ranges of SNRs. Introduction of domain knowledge (SNR) in the denoising block makes it less sensitive to noise. Hence it requires fewer data and less range of training SNR. We used a moderate variation of SNR [3, 5, 8, 12, 15, 20] dB for training. It is important to note, that MPR block, as a stand-alone network, needs to be trained on high SNR data. Otherwise, it will spend all its resources in removing noise. We do not concern ourselves with that since we train it with the output of the denoising block.

4.3.3 Dataset Specification and Computation Resources

We have simulated the OFDM packets based on IEEE 802.11 WiFi protocol using MATLAB which is a $N_f = 64$ subcarrier system. It is tested with practical indoor and outdoor urban channel models based on Winner II channel model [39] scenario B2 and ITU recommendations. Specifications of the channel conditions used for training are given in Table 4.1. Each packet is of length 5120 bits. LS estimate of the channel obtained from long preamble in frequency domain and received SNR are used as input to the denoising block as described in section 4.2.1. For this system, $N_{sc} = 52$. We have used the MMSE channel estimate as our training labels. The training set includes 12000 randomly generated training samples. We have tested our results with a test set including data for SNR ranging from 5 dB to 30 dB.

We have used TensorFlow [2] platform to train the model, with both stages trained for 500 epochs and a batch size of 128 each with a dynamic learning rate. The training starts with a learning rate of 0.001 and reduces by a factor of 0.8 every 50 epochs. This way gradient propagation slows down over time and reduces risk of gradient getting stuck at a saddle point. Training process is implemented on an Intel NUC (NUC7i7BNH) with i7-7567U processor and 16GB DDR4 memory, without any acceleration units like GPU or TPUs.

4.4 Complexity Analysis

In this section, we have analyzed the computational complexity of the proposed channel estimator. Given the proposed model will be trained offline, we are not concerned about the
Table 4.1: Different channel models used for experiment.

<table>
<thead>
<tr>
<th>Channel</th>
<th>Indoor</th>
<th>Outdoor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>Rayleigh (ITU recommendation)</td>
<td>Rayleigh (Winner II channel model scenario B2)</td>
</tr>
<tr>
<td>No. of taps</td>
<td>5</td>
<td>16</td>
</tr>
<tr>
<td>Path delays</td>
<td>[0 50 110 170 290 310] ns</td>
<td>[0 30 135 185 265 285 340 365 540 660 675 810 970 1340 170 290 310 340 365 540 660] ns</td>
</tr>
<tr>
<td>Doppler</td>
<td>0 Hz</td>
<td>3 Hz</td>
</tr>
</tbody>
</table>

denote the entire back-propagation algorithm for complexity analysis. We will focus on the computation complexity of channel estimation with the trained model, i.e., computing time complexity of the forward propagation of the network.

Complexity of a task or an algorithm is primarily defined by the maximum number of individual floating point operation or flops required to obtain the final output.

4.4.1 Denoising block

The encoder and decoder in the denoising block consist of only fully connected (FCNN) layers. The encoder network has three layers. The decoder network is a mirrored version of the encoder as shown in figure 4.1. So, we will analyze one of them and infer the other from there. In a fully connected neural network let’s denote number of neurons in $m^{th}$ layer as $n_m$. To obtain the output of $m^{th}$ layer, we have to compute a matrix multiplication and a subsequent non-linear transformation via the activation function. The shape of the input matrix of $m^{th}$ layer is $(N \times n_{m-1})$ and the weight matrix has the shape of $(n_{m-1} \times n_m)$, where $N$ is the number of datapoints, we wish to compute the prediction for. So the number of flops required to compute the matrix multiplications of $m^{th}$ layer is $(N.(2n_{m-1} - 1).n_m)$, out of which $(N.n_{m-1}.n_m)$ are multiplications and $(N.(n_{m-1} - 1).n_m)$ are additions. Moreover, for the activation function of $m^{th}$ layer, $(N.n_m)$ flops are required. So, if the network has $l$
layers, total number of flops required can be expressed as,

\[ n_{\text{Denoise}}^{\text{flops}} = \sum_{m=1}^{l} (N.(2n_{m-1} - 1)n_m) + (N.n_m) \]

\[ \approx \sum_{m=1}^{l} (N.(2n - 1)n) + (N.n) \quad \text{(generalizing)} \]

\[ = \sum_{m=1}^{l} N.(2n^2 + n) \]

\[ = l.N.(2n^2 + n) \quad \text{(4.6)} \]

In the encoder, \( l = 3 \) and in practice we predict channel frequency response for one packet at a time, so, effectively \( N = 1 \). Maximum number of flops required to compute the output of encoder network is,

\[ n_{\text{flops}} = 3.1.(2n^2 + n) \approx 6n^2 \quad \text{(4.7)} \]

So, the complexity of the encoder network and subsequently of the decoder network is \( \mathcal{O}(n^2) \). Since, in our implementation \( n \gg 6 \), so it does not incur any further cost in terms of order of magnitude.

### 4.4.2 MPR block

The MPR block consists of 1D convolutional layers and two fully connected layers. Complexity for fully connected layers can be obtained from equation 4.6. Time complexity analysis of a 1D convolution layer shows it requires \( \mathcal{O}(w.n.d^2) \) flops [65], where shape of each filter is \( w \times 1 \), \( n \) is the length of each channel and \( d \) is the depth along spatial dimension i.e, number of filters. Each filter is passed over the data along each data channel. Number of data channel is 2 for I and Q for input layer and \( d_k \) for \( k_{th} \) hidden layer.

In the MPR block, we have three convolution layers, where each layer has multiple filters. So, total number of flops required for one forward pass of a convolution layer of MPR
block is:

\[ n_{\text{flops}}^{\text{MPR}} = \sum_{i=2}^{4} (w.n_i.d_i^2) + (2w.n_1.d_1) \]  

(4.8)

where \( n_i \) is number of rows in each channel of input of \( i_{th} \) layer. In our application, the filter size is fairly low and does affect the number of computations is order of magnitude.

\[ n_{\text{flops}}^{\text{MPR}} \approx 3(n.d^2) + (2n.d) \]

where \( n_i \approx n, \quad d_i \approx d, \quad \forall i \)  

(4.9)

So, the computational complexity of MPR block can be given as:

\[ \text{Complexity}_{\text{MPR}} = \mathcal{O}(3(n.d^2) + (2n.d) + 2n^2) \]

\[ \approx \mathcal{O}(3(n.d^2) + 2n^2) \]  

(4.10)

We tend not to approximate this expression further, as for different number of subcarriers, \( n \) can change significantly. So, preemptively approximating further may lead to over/under-estimation of complexity.

### 4.5 Evaluation

We first investigate the individual performance of the different stages of the model. Then we evaluate the performance of the complete system in terms of mean squared error (MSE) curves in both indoor and urban outdoor channel conditions. Model performance is compared with least square (LS) method, minimum mean squared error (MMSE) method, practical MMSE (described in section ??), fully connected networks - blackbox model (FCNN1, FCNN2), Denoising block, Multipath resolution (MPR) block and complete two-stage domain knowledge aided model (DK-model).
Table 4.2: Training data used in different networks.

<table>
<thead>
<tr>
<th>Network</th>
<th>Denoising block</th>
<th>FCNN1 (all SNR)</th>
<th>FCNN2 (selective SNR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training data</td>
<td>12000</td>
<td>30000</td>
<td>12000</td>
</tr>
<tr>
<td>Test data</td>
<td>2000</td>
<td>2000</td>
<td>2000</td>
</tr>
<tr>
<td>SNR range for training data</td>
<td>3, 5, 8, 12, 15, 20 dB</td>
<td>5-30 dB</td>
<td>3, 5, 8, 12, 15, 20 dB</td>
</tr>
</tbody>
</table>

### 4.5.1 Denoising Block

![Figure 4.2](#)

(a) Variance L.  
(b) Variance Z.

Figure 4.2: Comparing how noise sensitivity decreases in the sample \( Z \) from latent space \( L \).

![Figure 4.3](#)

(a) Accuracy.  
(b) Loss.

Figure 4.3: Performance of Denoising Block against generic feed forward network.

1) **Effect of modified variance of latent vectors**: As described in section 4.2, variance of the latent variable is updated based on equation 4.1. Figure 4.2 shows the change in variance of the latent space before denoising for \( L \) and \( Z \) using the modified variance term after
denoising. It indicates that due to the noise effect, variance decreases with increasing SNR, but the modified model generates variance of latent space independent of SNR.

2) Model performance: Figure 4.3 compares the training performance of the denoising block with a blackbox approach using fully connected neural networks. The FCNN contains two hidden layers and 208 neurons in each layer. It is trained in two different scenarios, 1) FCNN1 - trained with data from signals transmitted at SNR range (5-30 dB), 2) FCNN2 - trained with data from selective SNR ranges, same as used for the denoising block. The dataset specifications for all three networks are given in table 4.1. As we see in Figures 4.4a and 4.4b, all three networks perform similarly at low SNR. At higher SNR, denoising block shows about 6 dB improvement over LS estimate and 1.8 dB over FCNN1. FCNN1 however shows 6 dB improvement over FCNN2. So, FCNN1 reaches comparable accuracy to denoising block with 30000 training samples, whereas denoising block reached the same accuracy with 60% fewer samples (12000). This emphasizes that domain knowledge aided design requires less training data to reach better accuracy.

3) Training latency: Latency, in this context, is defined as the number of epochs required to reach a certain performance metric for the model, assuming each epoch requires almost equal computation time. FCNN1 has higher unit computation time as its training set is larger than that of denoising block, given same computation resources. The FCNN1 reaches the MSE performance at 500 epochs, whereas the denoising block reaches to that level of MSE shown in Figure 4.4a in 90 epochs. This demonstrates that the denoising block reaches higher accuracy with lower latency.

4.5.2 Multipath Resolution Block

Performance of the multipath resolution block is evaluated by comparing its performance with that of the legacy methods and denoising block. MPR block is trained on both denoised data, output of denoising block, and noisy input with same dataset that is used in denoising block for system evaluation.

2) MSE performance: Figure 4.5 shows the MSE performance of the MPR block trained on noisy data in both indoor and outdoor channels along with the performance on denoised data. The MPR block shows 6.2 dB and 5.5 dB improvement over LS estimate in indoor and outdoor channels respectively. This is not achievable from denoising block as its capacity
Figure 4.4: Performance comparison of denoising block with respect to fully connected network with training data belonging to all SNR ranges and selective SNR range.

(a) Indoor channel.  
(b) Outdoor channel.

is restricted to noise removal only. We observe that at high SNR, MPR block outperforms denoising block by 5.3 dB in outdoor channel. In indoor channel, we observe 1.2 dB improvement as the multipath effect is mostly mitigated by guard band.

4.5.3 Full domain knowledge guided model evaluation

Figure 4.5 shows the performance of the proposed model in terms of MSE in comparison to the denoising block, MPR block, LS, MMSE, and practical MMSE. In indoor channel (Figure 4.5a), DK-model shows 8.8 dB improvement over LS and 5.9 dB improvement over
practical MMSE. At low SNR, DK-model has $\sim 1.1$ dB improvement over both individual denoising block and MPR block, while at high SNR, it outperforms denoising block by 3.3 dB and MPR block by 2.4 dB. In outdoor channel as shown in Figure 4.5b, DK-model shows 9.3 dB improvement over LS and 5.9 dB over practical MMSE. At low SNR, DK-model shows an improvement of 2.1 dB over both denoising block and MPR block, whereas, at high SNR, we observe 8.2 dB and 2.9 dB improvement over denoising and MPR blocks respectively.

4.6 Conclusion

We have developed a two-step wireless channel estimation technique powered by both wireless communication domain knowledge and deep neural networks. This work addresses the issue of additive noise and strong frequency selectivity experienced by most modern day wireless systems. These findings can advance the design of estimation systems, deployable in a practical receiver.
CHAPTER 5
A Case for OFDM in Ultrabroadband Terahertz Communication:
An Experimental Approach

Terahertz (THz) Band communication is envisioned as one of the leading technology to meet the exponentially growing data rate requirements of emerging and future wireless communication networks. Although prior works on THz channel measurement have observed strong large scale fading effects, like free space path loss and atmospheric absorption, they have not shown the overall effect on the received signal due to additional non-linearity from experimental hardware. In this work, we perform over the air experiments and channel measurements in ultra-broad THz bandwidth (10 GHz), much wider compared to prior testbed setups [3,60], which reveals that the received signal undergoes frequency selective fading due to hardware imperfections. This motivates us to utilize orthogonal frequency division multiplexing (OFDM) in indoor short range ultra-broadband THz communication. To experimentally show its efficacy, we compute the coherence bandwidth of the system, which is less than the full bandwidth. We derive optimal parameters for OFDM for our hardware setup and perform over-the-air experiments to achieve 42 Gbps physical layer data rate over an effective bandwidth of 8.125 GHz.

5.1 Contribution

Here, we summarize our contributions as follows:

1. **Attenuation Model for THz**: We have developed an attenuation model for THz band that incorporates frequency dependent hardware impairments along with the large scaling fading effects.

2. **New Waveform Development**: We have modeled OFDM parameters required to tackle hardware impairments and generated OFDM waveform with these parameters for our testbed.

3. **Over the Air Experiments with Designed Waveform**: We performed over-the-air experiments in an indoor setup with the OFDM waveform we have developed for different modu-
5.2 A Case for OFDM

In the THz frequencies, multipath effect (i.e., small scale fading) is negligible compared to large scale fading, primarily caused by free-space pathloss and atmospheric absorption loss. Figure 5.1a presents the atmospheric absorption at THz frequency caused by primarily oxygen and water vapor based on the guidelines provided by ITU model [30]. We observe severe attenuation peaks at some frequencies which are likely to be no transmission zones. The extent of free space pathloss (FSPL) is shown in Figure 5.1b at different frequencies and distances. Hence, the channel can be assumed as a flat fading channel. Based on this, the fading component, $\alpha$, can be expressed as a combination of path loss, $L_{\text{path}}$ and absorption loss, $L_{\text{absorption}}$.

$$\frac{1}{\alpha(f, d)} = L_{\text{path}}(f, d) * L_{\text{absorption}}(f, d).$$  \hspace{1cm} (5.1)

These components are dependent on carrier frequency $f$ and distance $d$ between the transmitter and receiver. To operate in wider bands $B$, like 10 GHz, we need to know the frequency response of the channel. We define the attenuation over a bandwidth of $B$ as:

$$\Delta \text{Loss}(f, d) = 10 \log_{10}[[1/\alpha(f + B/2, d)]/[1/\alpha(f - B/2, d)]]$$  \hspace{1cm} (5.2)
Figure 5.2: The difference of attenuation over 10 GHz band at different transmit frequencies and distances. Figure 5.3: Power Spectral Density of single carrier waveform in a bandwidth of 10GHz.

Figure 5.2 shows this difference $\Delta Loss(f, d)$ over a bandwidth of $B = 10 \ GHz$ at different transmit frequencies $f$, and shorter distances $d$ owing to the combined effect of FSPL and atmospheric absorption. From this theoretical analysis, other than the absorption lines at around 180 GHz and 320 GHz, the channel can be considered flat (difference of $\sim 0.5 \ dB$) at any given distance and transmit frequency.
5.2.1 Effect of Transceiver Hardware

As we venture towards wider bandwidths to utilize the available spectrum for communication, it is expected that frequency response of the transmitter and receiver hardware will not remain flat \[19,66\]. Since, the hardware used in terahertz testbeds is mostly experimental and hosts a plethora of different non-linearity inducing components, the transmitted and received signals consist of significant frequency dependent hardware impairments, which are specific to that testbed. This loss includes but is not limited to hardware dependant phase noise, frequency dependent antenna gain, and other combinations of various circuits where practical filters have different effects on signal at different frequencies. So, we denote the transmitter hardware loss, \( L_{hw}(f) \), as a frequency dependent loss and update the initial loss expression (equation 5.1) as:

\[
\frac{1}{\alpha(f, d)} = L_{\text{path}}(f, d) \ast L_{\text{absorption}}(f, d) \ast L_{hw}(f).
\] (5.3)

Effect of hardware impairment and resulting attenuation causes the received signal to have variation in channel gains over bandwidths of several gigahertz. It is evident from Figure 5.3, where attenuation across a 10 GHz band at center frequency 140 GHz shows \( \sim 7 \) dB attenuation opposed to \(< 1 \) dB from the theoretical calculation from ITU recommended propagation and absorption loss model described before. This proves the fact that ultra-broad bands at THz frequency has significant frequency selective fading. Hence, it is essential to model the coherence bandwidth to analyze the effect of hardware on signal reception.

5.2.2 Coherence Bandwidth

Coherence bandwidth \( (B_c) \) is a statistical measurement of the range of frequencies over which the channel can be considered flat. In other words, it is a measure of the approximate maximum bandwidth or frequency interval within which two frequencies of a signal are likely to experience comparable or correlated amplitude fading. Coherence bandwidth of frequency selective channel can be calculated based on channel correlation peaks. We define the frequency domain channel response as \( H \), which captures not only the effects of wireless channel but also the combined effects of hardware impairments of both transmitter and receiver. Channel correlation, in a generalized term, is defined as the time-frequency
correlation of channel coefficient $R_{HH}(t + \Delta t, f + \Delta f)$:

$$ R_{HH}(t + \Delta t, f + \Delta f) = E[H(t, f)H^*(t + \Delta t, f + \Delta f)] $$

The channel is assumed to not change over time and includes minimal to no path delay as multipath effect is nominal. Thus, the time-frequency correlation term boils down to frequency correlation function $R_{HH}(0, \Delta f)$. The coherence bandwidth, for a given correlation coefficient $\gamma_f \in [0, 1]$, is defined based on channel frequency correlation as given in equation (5.5):

$$ B_c = \frac{1}{2} \left[ \max_{\Delta f > 0} \left( \frac{|R_{HH}(0, \Delta f)|}{R_{HH}(0, 0)} = \gamma_f \right) - \min_{\Delta f < 0} \left( \frac{|R_{HH}(0, \Delta f)|}{R_{HH}(0, 0)} = \gamma_f \right) \right] $$

We can choose $\gamma_f$ based on the system requirement, where higher $\gamma_f$ denotes a smaller coherence bandwidth and a smaller $\gamma_f$ denotes a larger coherence bandwidth. Based on $\gamma_f$ requirement, if $B_c$ is smaller than the total transmission bandwidth $B$, the channel experiences different fading effects across the transmission bandwidth. This creates frequency selective fading. Conventional single-carrier waveforms may require complex equalization schemes to combat frequency-selective fading. The ideal equalizer should have a frequency response that is the exact inverse of that of the channel. This necessitates an infinite number of equalizer taps, which is impractical. Furthermore, noise gets enhanced in the equalization process making the link unusable in a deep fade.

### 5.2.3 OFDM as a Candidate Waveform

Orthogonal Frequency Division Multiplexing (OFDM) is widely used as the primary solution for frequency selective fading caused by multipath. It distributes the entire channel into smaller orthogonal subcarriers. Channel within individual subcarrier behaves as flat narrowband channel solving the frequency selective fading. Therefore, we find OFDM waveform suitable to combat the hardware induced frequency selectivity common in wide-band systems as well. However, the effect of multipath, for which OFDM was designed, is different from the effect of hardware generated distortion. In the following subsections, we carefully analyze and design the three main parameters of OFDM – subcarrier width, FFT length,
and cyclic prefix length tailored to the system under consideration.

5.2.3.1 Subcarrier Width

In OFDM, the number of subcarriers is selected such that each of their bandwidths is smaller than the coherence bandwidth. This ensures that each subcarrier undergoes flat fading, which can be equalized individually at the receiver. In other words, if $f_{sc}$ is the subcarrier width, the necessary condition is $f_{sc} \leq B_c$. Additionally, the subcarrier width needs to be such that an integer number of subcarriers are contained within 1 symbol period. Thus we can define the conditions of selecting subcarrier width as: $\exists f_{sc} \leq B_c : k \cdot f_{sc} = \frac{1}{t_{sym}}$, $k \in \mathbb{Z}^+$, where $t_{sym}$ is one symbol period. For simplicity, we will use $k = 1$ here onwards.

5.2.3.2 FFT length

To exploit the computational efficiency of Discrete Fourier Transform (DFT) with FFT length equal to $2^n$, $n \in \mathbb{Z}^+$, we need number of subcarriers = $2^n$ as well. Thus to satisfy the previous two conditions, the minimum FFT length needs to be equal to the least integral power of 2, higher than $B/f_{sc} \Rightarrow$ higher than $B/B_c$, according to the boundary condition of $f_{sc}$. The FFT length and subsequently minimum number of subcarriers $N_{sc}$ can be derived as:

$$N_{sc} = 2^{\lceil \log_2(B/B_c) \rceil} \quad (5.6)$$

The updated subcarrier width is $f_{sc} = B/N_{sc}$.

5.2.3.3 Cyclic Prefix

The purpose of cyclic prefix (CP) is to eliminate inter-symbol interference (ISI) and inter-carrier interference (ICI) introduced due to multipath. So, it may seem that CP is unnecessary in absence of multipath, which will improve utilization of the bandwidth. However, it also aids in synchronization, reducing error in packet detection. The amount of CP necessary for our hardware setup is derived empirically in section 5.5.
5.3 Derivation of OFDM Parameters

In this section, we derive the parameters of OFDM specific to our testbed setup described in section 5.4.2. Figure 5.4 shows the frequency correlation across different frequency shift $\Delta f$ and corresponding coherence bandwidth for correlation coefficient $\gamma_f = 0.8, 0.85, 0.9$ for the 10 GHz wide channel centered at 140 GHz. In most applications, practical value of correlation coefficient ranges from $0.5$ to $0.9$ [24]. We opt for the maximum of that, $\gamma_f = 0.9$. Using equation 5.5, we get $B_c = 468.6$ MHz for $\gamma_f = 0.9$. Using the value of $B_c$ and available transmission bandwidth of $B$ (10 GHz) in equation 5.6, we get the minimum number of subcarrier $N_{sc} = 32$ and corresponding subcarrier width $f_{sc} = 312.5$ MHz. Instead of creating an OFDM packet structure with 32 subcarriers, we utilize a well-defined OFDM structure of IEEE 802.11 standard [29] with 64-pt FFT that also meets the requirement above. It provides the preamble structure that can be used for packet detection, carrier frequency offset correction, and initial channel estimation. The pilots are inserted in the packets based on the same protocol for further channel tracking and residual phase offset correction. With 64 subcarriers and a bandwidth of 10 GHz, subcarrier width $f_{sc}$ is 156.25 MHz and the corresponding symbol duration becomes $t_{sym} = 6.4$ nanoseconds.
5.4 Implementation Details

The software side of implementation in our work can be categorized in two sections: 1) OFDM baseband signal generation (Tx) and post-processing (Rx), and 2) testbed specific modifications. Figure 5.5 provides a concise block diagram representation of both baseband signal generation setup in software and the testbed. Additional details of these are provided below.

5.4.1 Baseband OFDM:

Transmitter: The input bitstream is modulated based on four baseband modulation techniques: BPSK, QPSK, QAM16, and QAM64, followed by pilot insertion at specified locations. These symbols are then converted to time domain using inverse fast Fourier transformation (IFFT) and cyclic prefix is appended to each symbol. Finally, short and long preambles are added at the beginning of the signal to create a packet ready for transmission.

Receiver: Packet detection is performed by cross-correlating the received signal with long preamble. Then, the carrier frequency offset is estimated and compensated, followed by removal of CP. Subsequently, fast Fourier transform (FFT) is performed to retrieve frequency domain symbols. Channel estimation and equalization are performed using least square estimate based on long preamble. Finally, the pilot tones are used to remove any residual phase offset and the symbols are demodulated by hard decision decoding to bit stream.
5.4.2 Testbed Overview

Our advanced testbed setup at Air Force Research Laboratory (AFRL) in Rome, New York, consists of state-of-the-art equipment specially designed for operating in ultra-broad bandwidth (10 GHz) over a wireless link in 140 GHz center frequency. The testbed is shown in Figure 5.6. At the transmitter, a Keysight E8257D Performance Signal Generator (PSG) functions as the local oscillator (LO) source at a frequency of 11.67 GHz. A custom, Schottky-diode based Mixer/Amplifier/Multiplier Chain (MixAMC) upconverts the LO to the operating frequency of 140 GHz with a $\times 12$ multiplier chain. The MixAMC mixes an intermediate frequency (IF) signal from the Keysight M8194A Arbitrary Waveform Generator (AWG). This cutting-edge device can sample up to 120 gigasamples per second (GSa/s) of analog bandwidth of up to 32 GHz with 8-bit resolution for up to 520,000 data samples. The AWG can be set to output voltage levels anywhere between 75 mV and 800 mV. The maximum IF power going into the mixer is 0 dBm and the maximum output power from the transmitter is -15 dBm (30 $\mu$W). Directional, conical horn antennas from VDI are used for transmission and reception of the signal. These antennas have 20 dB gain and 13° half-power beamwidth.

On the receiver side, another PSG sets the LO and a VDI MixAMC down-converts the received RF signal to the IF band. From here, the received signal is passed to the state-of-the-art Keysight DSOZ632A Digital Storage Oscilloscope (DSO). Sampling at 160 GSa/s,
we can view both the frequency and time-domain signals and capture up to 63 GHz of real-time bandwidth for offline processing. Because of the high sampling rates and equipment bandwidth capacity, we can transmit ultra-broadband OFDM signals at 140 GHz. With this setup, we load our custom-designed OFDM waveforms onto the AWG where it is sampled and sent through the upconverter. We can vary the voltage setting of the AWG as well as the distance between the transmitter and receiver sides to examine the OFDM wireless transmissions under different conditions. From there, we capture the received signals with the DSO and analyze the results in MATLAB.

5.4.3 Testbed Specific Modifications:

Transmitter: The sampling rate of AWG is fixed at 120 GSa/s and the frontend bandwidth is 10 GHz, which necessitates upsampling the baseband signal 12 times to match the sampling rate. After that, the signal is passed through a pulse shaping filter and passband modulated to obtain the real valued passband signal. Then, the signal is scaled to fit the dynamic range of AWG. These steps are required to match testbed requirements. Receiver: The passband signal is demodulated to get the complex envelope of baseband back. The sampling frequency of DSO is 160 GSa/s, for which the received signal is downsampled to meet the nominal sampling rate (Nyquist rate based on signal bandwidth without oversampling) of 10 GSa/s. Finally, the signal is passed through a pulse-shaping filter to retrieve the baseband signal back for standard OFDM receiver processing in the receiver block.

5.5 Experimental Results

Here we present the experimental results of signals transmitted and received at 140 GHz center frequency and 10 GHz bandwidth. We examine the effect of different length of cyclic prefix and evaluate error vector magnitude (EVM) and bit error rate (BER). We choose three CP lengths for inspection in our experiments: 0, 4 (1/16th symbol), and 16 (1/4th symbol) samples. The WLAN standard [29] uses 1/4th symbol as a CP, which is required to address the multipath in an indoor environment. In the THz band, where multipath is not the main cause for frequency selectivity, 16 samples might be redundant. This is why we choose 4, which provides fewer samples to address any synchronization errors. This choice
Figure 5.7: Error Vector Magnitude over Transmitter (AWG) amplitude for different guard length.

has less overhead and better utilization of channel compared to 16 samples.

To understand whether any guard is required or not, we also perform experiments with no guard. We designed a series of experiments where we collect signals for all the modulation orders (BPSK, QPSK, QAM 16, QAM 64) for each CP length (0, 4, 16). By changing both the output voltage level (100 mV to 600 mV) of the AWG and the distance (10 cm to 50 cm) between the transmitter and receiver, we created different propagation conditions to test our proposed waveform in over the air transmission and reception. We have transmitted OFDM packets generated with five different seeds, each consisting of 20 OFDM symbols for each of these scenarios. Each capture at the receiver contains 5-7 packets with 3 instances of capture with identical transmit-receive conditions. So, total number of captured packets for each choice of cyclic prefix length is 1800, i.e, 36000 OFDM symbols.

**Error Vector Magnitude (EVM):** EVM denotes the absolute distance of a received symbol from its expected value, thus providing a direct measure of symbol error. To examine only the effect of the lengths of CP on the received signal, we fixed the distance between transmitter and receiver at 10 cm, which in turn also removes any effect of manually moving the transmitter-receiver pair and antenna realignment issues. We compared the EVM of all different modulation orders at different AWG voltages as shown in figure 5.7. We observed higher EVM for signals without any CP compared to the two scenarios with CP. We also found that CP lengths of 4 and 16 perform similarly with varying AWG voltages. This reaffirms our expectations, as discussed in section 5.2 and section 5.3 that there is practical benefit of using CP even in minimal multipath scenarios. Therefore, we conclude that 1/16th
FFT length is an adequate CP length for the testbed under consideration.

**Bit Error Rate (BER):** Figure 5.8 demonstrates the bit error rate (BER) achieved over received SNR for different modulation orders and cyclic prefix lengths. Consistent with the trends in EVM, we notice that CP lengths 4 and 16 have very similar performances in BER. Without any CP, the BER does not reduce at higher SNRs ending up with sub-optimal performance. For instance, in case of QAM 16 modulation, at received SNR of 26 dB, BER values, averaged over all packets, are $1 \times 10^{-3}$, $1.19 \times 10^{-3}$ and $3.6 \times 10^{-3}$ for CP lengths 16, 4 and 0 respectively. As evident in Figure 5.8, BER for higher modulation orders is not improved significantly. Analysis of this is out of the scope of this work and are evaluated in future work.

**Physical Layer Rate:** Based on the symbol duration $t_{sym} = 6.4$ ns, as described in, section 5.3, we can estimate physical layer data rate of our system in bits per second (bps). If modulation bits per OFDM symbol is denoted as $N_{dbps}$, CP length is $N_{CP}$, FFT length = $N_{FFT}$, then data rate (D) can be expressed as: $D = \left[ t_{sym} \left( 1 + N_{CP}/N_{FFT} \right) / N_{dbps} \right]^{-1}$. Table 5.1 presents the achievable data rate for combinations of different modulation order and CP length. A maximum datarate of 42.3528 Gbps is achieved with CP length of 4 and

### Table 5.1: Physical layer data rate at different modulation and $N_{CP}$

<table>
<thead>
<tr>
<th>Modulation order</th>
<th>Bits per symbol ($N_{dbps}$)</th>
<th>Data rate (in Gbps) $N_{CP} = 4$</th>
<th>Data rate (in Gbps) $N_{CP} = 16$</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPSK</td>
<td>48</td>
<td>7.0588</td>
<td>6</td>
</tr>
<tr>
<td>QPSK</td>
<td>96</td>
<td>14.1176</td>
<td>12</td>
</tr>
<tr>
<td>QAM 16</td>
<td>192</td>
<td>28.2352</td>
<td>24</td>
</tr>
<tr>
<td>QAM 64</td>
<td>288</td>
<td>42.3528</td>
<td>36</td>
</tr>
</tbody>
</table>
the highest modulation order QAM 64.

5.6 Conclusion

We have experimentally shown the effect of hardware impairments on THz band wireless signal. We have developed a testbed and performed over the air THz band transmission to verify how OFDM improves performance by mitigating various shortcomings of the THz spectrum, namely: frequency selective fading due to hardware impairments and synchronization error, thus making a case for OFDM on an experimental basis.
CHAPTER 6
Communication Knowledge Aided Neural Network for OFDM Receiver in Terahertz Band

Ultra-broadband communication in emerging spectra, like Terahertz (THz) band, is the frontier to meet the data rate requirements of future wireless communication systems. Existing signal processing based methods are developed for sub-6 GHz band, which cannot capture the intricacies in ultra-broad THz bandwidth and non-linearities arising from hardware. Development of additional testbeds for the THz band is also limited by lack of physical layer components well equipped to tackle the issues present in ultra-broad terahertz band. To overcome these limitations, we develop neural network (NN) models for OFDM receiver in ultra-broadband, where expert knowledge of wireless communication is infused in different stages and parameters of the model to create a practical receiver that can adapt to different wireless environments. The parameters of the NN are derived from the underlying theory and can be adapted to different wireless environments. Our model is designed to capture the correlation between real and imaginary components of wireless signals, that can be trained with limited data. The models are trained with over-the-air captured OFDM signals, transmitted in the THz band with 10 GHz bandwidth. Our results show significant improvement in bit error rate (BER) for different modulation orders (upto 6 dB in BPSK and 1.2 dB in QAM 64) compared to existing signal processing based receiver designs.

6.1 Contribution

The contributions of this work can be summarized as:

1. We have designed a NN based receiver for OFDM waveform in ultra-broad THz band, where the domain knowledge guides the model design to tackle the errors in complex signals and minimizes data required for training.

2. We have injected parameters in the model that can be adapted to any wideband multicarrier system and the NN model parameters can be derived according to the specific properties
of the communication protocol.

3. We developed a **multi-stage NN**, where the first stage operates in real domain to improve distortions in channel estimation and the second stage is composed of a set of parallel identical NNs that operates in complex domain to reduce error in equalization.

4. We introduce a complex activation function, Leaky Complex ReLU (LCReLU) to compensate for the gradients in negative regions of complex wireless signal samples.

5. We trained the NN model with **over-the-air signals** for different modulation orders that are captured from an indoor THz testbed with 10 GHz bandwidth.

6. Our results show up to 6 dB and 3.8 dB **improvement in BER** over least square (LS) and approximate LMMSE channel estimation based receiver designs respectively.

To the best of our knowledge, **this work is the first** to implement NN based receiver design a) in over the air transmitted OFDM packets and b) in practical testbed for THz communication.

### 6.2 Physical Layer Receiver Design

In the Received wireless signal at THz band, distortions are present caused by additive Gaussian noise, propagation conditions present in wireless channel and hardware distortions caused by the practical hardware in use for transmission-reception. To mitigate these errors at the receiver, we develop two stages of neural network models: 1) Refined Channel Estimation (RCE) block and 2) Enhanced Equalization (EE) block. Figure 6.1 shows the
complete block diagram of our receiver, where the blue blocks are DSP based and the green blocks are NN based designs.

6.2.1 Refined Channel Estimation (RCE) Block

In practical receivers, channel is estimated using least square (LS) estimation technique from synchronization symbols or preambles. Additive white Gaussian noise (AWGN) and other distortions caused by wireless channel are retained in the LS channel estimate, which is propagated to the equalized symbols and to information bits. To alleviate this distortion, we introduce a variational autoencoder (VAE) \[37\] that converts the channel estimate to a lower dimensional latent space, removes the noise variance based on the SNR and reconstructs the denoised signal back. As noise does not induce in-phase (I) and quadrature-phase (Q) sample correlation, they can be dealt with separately. Thus, we have modeled the VAE in real domain, where real and imaginary parts are stacked one after another.

The VAE in the RCE block is shown in figure 6.1, where the input is the LS channel estimate $H_{LS}$. The width of the input is twice the number of non-empty subcarriers to incorporate real and imaginary components. The inputs are converted to a lower dimensional latent space $L$, keeping their features intact along with the distortion, which primarily consists of zero mean additive white Gaussian noise (AWGN) $\sim \mathcal{N}(0, \sigma_n^2)$. Noise, being independent of the signal, only modifies the variance of the latent space $\sigma_L^2$ and is a linear combination of the noise variance and latent space variance. We get rid of the noise variance from $\sigma_L^2$ using signal to noise ratio (SNR) as $\text{SNR} \propto 1/\sigma_n^2$. We get the updated variance $\sigma_Z^2 = \sigma_L^2/\text{SNR}$. $\mu_L$ and $\sigma_Z^2$ are used to generate the sampled space $Z$. The decoder being a generative network converts the sampled space $Z$ to rectified output $\hat{H}$. Since, estimated SNR from same receiver hardware is being used in both training and validation stages of the neural network, effect of estimation error will be similar in both stages and no additional calibration is required to prevent the effect of estimation error.

The objective function of this model $\text{Loss}_{RCE}(\theta, \phi)$ is:

$$\text{Loss}_{RCE}(\theta, \phi) = - \sum_{i=1}^{N} \mathbb{E}[\log(p_{\theta}(\hat{H}_i|Z))] + \lambda KL(q_\phi(L|H^L_{i}|)||p(L))$$

(6.1)
The first term in (6.1) is the decoder objective, which represents our primary goal to minimize the negative log-likelihood between the sampled space representation and the rectified channel estimate. The second term is the encoder objective, which minimizes the Kullback-Leibler (KL) divergence between the approximate posterior $q_\phi(L|H_{LS}^i)$ and the latent distribution $p(L)$. It captures the information loss due to encoding into a lower dimensional space and regularizes the primary (decoder) objective by preventing the encoder to generate discrete latent representation. We added a positive scaling term $\lambda (\ll 1)$ to this, because even though regularizers keep the training in check from overfitting, stronger influence of this may end up masking effects of different SNRs on the signal latent space and the decoder output will end up with more noise. Both the encoder and decoder networks are fully connected (FCNN) with three layers each. We have chosen Leaky Rectified linear unit (LReLU) as the non-linear activation function for all the layers except the final decoder output, which is linear. We added a leak of 30% to provide access to the negative I and Q inputs for the gradient to propagate efficiently. In chapter 4 we have discussed about developing a similar approach to reduce the noise in channel estimate that has been implemented in our prior work [10].

The fundamental difference compared to the previous approach is that 1) we regenerate the improved channel estimate $\hat{H}$ in an unsupervised manner as we do not have the true channel state Information in a wireless testbed and 2) we use an additional regularizer ($\lambda$) term to keep the encoder influence in check on the decoder objective, which was not necessary due to the supervised nature of decoder network in the previous approach.

Figure 6.2: A case study for training data requirement.
6.2.2 Enhanced Equalization (EE) Block

**Need for Complex NN**: Ultra-broadband wireless signals contain distortions that may arise from hardware non-linearity at both the transmitter and receiver ends. These distortions often correlated in real and imaginary components inducing I, Q correlation in received symbols [17]. Since the characteristics of this distortion are not known at receiver side, it cannot be removed using channel estimation techniques derived for known channel distributions e.g., Rayleigh, Rician, etc. So, we explore the feature space of the received signal in complex domain by embracing complex neural network [63] to remove such errors in equalization. This provides us a unique advantage to explore effects of I and Q correlation. Additionally, it provides higher degrees of freedom for the gradient to propagate ensuring better optimization performance.

**Enormous Data Requirement**: It is difficult to quantize the baseline requirement of training samples due to its subjective nature leaning on data type and the objective function itself. However, if there are $F$ features in a data set, multiple different samples are required for each feature to generalize the NN model [4]. Based on that, as OFDM is modulated in frequency domain, the minimum number of required samples is in the order of $M^{N_d}$, where $M$ is the modulation order and $N_d$ is the number of data subcarriers. Consider a simple OFDM
structure with 4 subcarriers and QPSK modulated data as shown in Figure 6.2. This leads to number of unique OFDM symbols $= M^{N_d} = 64$. The minimum number of data samples required for training without any variation in individual features is of this order. Following this, in case of a 64-pt FFT and the lowest modulation order (BPSK), the minimum number of data samples required for training is $2^{64} \approx 2 \times 10^{19}$. Keeping in mind, affinity of current wireless systems towards higher modulation orders and larger FFT sizes, collecting such an enormous amount of data and training with those are infeasible tasks in practice.

To alleviate this issue, we introduce correlated subcarrier group, $k$. The idea originates from the fact that even though the OFDM subcarriers are orthogonal to each other representing a set of orthogonal bases or features for the OFDM symbols, there is persistent correlation among the sets of closely packed subcarriers when the symbols propagate through frequency selective media. This correlation is primarily driven by coherence bandwidth of the channel which is defined as the frequency band across which the channel can be considered flat. Thus, $k$ subcarriers within a coherence band are expected to be correlated. This can be derived as $k = ([B_c/w_{sc}])$, where $B_c$ is the coherence bandwidth of the channel and $w_{sc}$ is the width of individual subcarrier. From the perspective of NN architecture, $k$ is a group of subcarriers, where the features are correlated within a group but has little to no correlation across groups. Thus, each group can be treated independently to explore their feature space to minimize hardware induced non-linear distortions and correlation among inter-group features can be skipped without significant information loss. Hence, we propose to design EE block consisting of $N_{net} = N_d/k$ units of parallel identical fully connected NNs, each operating on a group of $k$ consecutive subcarriers. Figure 6.3 summarizes the symbol splitting based on subcarrier correlation and development of EE block architecture in a generalized manner. Splitting the input symbols drives the number of unique symbols down to $M^k$, which is orders of magnitude less than $M^{N_d}$. Thus the baseline requirement of training dataset size is also reduced. Input to the EE block is one tap equalized OFDM symbol, $Y_{eq}$, which is split into $N_{net}$ units. The final output of these units are concatenated to generate the output symbol $\hat{Y}$. Since this is treated as a regression problem, the objective for each network $(i)$ is to minimize the mean squared error between the subcarrier entries of
the network of interest and its labels given as:

$$\text{Loss}^i_{EE} = \sum_{m=1}^{k} \| \hat{Y}^i_m - X^i_m \|^2, \quad (6.2)$$

where $\text{Loss}^i_{EE}$ is the objective function of the $i^{th}$ network and $\hat{Y}^i_m$ and $X^i_m$ are the $m^{th}$ output and label respectively of the $i^{th}$ network.

EE block with architectural specifications and hyperparametric details is shown in figure 6.1 which results from a practical case study presented in section 6.3. Each EE unit is designed as four layers of feed-forward complex network. We define a complex activation function, Leaky Complex ReLU (LCReLU), as

$$\text{LCReLU}(Z) = \text{LReLU}(R(Z)) + i\text{LReLU}(I(Z)), \quad (6.3)$$

$$\text{LReLU}(x) = \begin{cases} \alpha x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$$

This is similar to the CReLU activation function, which performs ReLU operation on real and imaginary components. CReLU performs ReLU operation independently on real and imaginary components of the output of the neuron. We are able to do that, as, for a complex symbol, phase is unrestricted with a range of $[0, 2\pi]$. But this restricts the gradients to positive real and imaginary components only. We introduced a 30% leak in the CReLU function as I and Q components can not be restricted to positive values.

### 6.3 OFDM Receiver in THz - A Case Study

In chapter 5 we developed OFDM signal parameters for a 10 GHz wide THz band indoor communication with experimental verification. Here we will derive and specify the neural network parameters of the proposed receiver design and evaluate the system against the state of the art. Figure 6.4 provides an overview of the complete system block including the software and testbed components along with the exact position of the proposed receiver design in the full-stack transceiver chain. In the 10 GHz system, we analytically calculated the coherence bandwidth $B_c = 468.6$ MHz and subcarrier width $w_{sc} = 156.5$ MHz with 64 subcarriers. We have 48 data subcarriers and 4 pilot subcarriers. Other 12 subcarriers are used as guard and left empty. The THz testbed specific design parameters are also shown
**Figure 6.4**: Block diagram presenting different components of the Implementation in figure 6.1.

**RCE Block**: The input to the block is LS channel estimate derived from preamble. There are 52 nonempty subcarriers with I and Q samples concatenated making the input dimension $104 \times 1$. The encoder network has 3 layers with 208, 52 and 26 neurons respectively in each layer. The latent vector is of shape $26 \times 1$. The sampled space also has a shape $26 \times 1$. The decoder mirrors the encoder network. It has 3 layers with 52, 208 and 104 neurons in each layer, yielding the output shape to be $104 \times 1$.

**EE Block**: The input to the EE block is equalized data symbol with 48 subcarriers. Given the coherence bandwidth and subcarrier width of our system, the coherent subcarrier group, $k = 4$, and number of NNs, $N_{net} = 12$ are derived. Thus, 12 NNs have been instantiated, each of them has a $4 \times 1$ complex tensor as input. Each network consists of 4 layers with 8, 32, 16, 4 neurons respectively. The outputs of these 12 NNs are concatenated to form an equalized OFDM symbol with 48 data subcarriers.

### 6.4 Experimental Setup

#### 6.4.1 Baseband Processing

We have generated baseband signals for four different modulation orders: BPSK, QPSK, QAM16 and QAM64. The baseband signal generation with testbed specific modification for over the air transmission and post reception baseband processing are all performed in MATLAB. The signal generation steps are as follows: information bit generation, mod-
ulation, pilot insertion, IFFT, addition of cyclic prefix (CP) and insertion of preambles to generate time domain baseband signal. The receiver processing steps are as follows: detection of the start of packet, carrier frequency offset (CFO) compensation, CP removal, FFT, channel estimation, equalization, phase error correction, and demodulation to retrieve the information bits.

6.4.2 THz Testbed

Our advanced testbed setup at Air Force Research Laboratory (AFRL) in Rome, New York, consists of state-of-the-art equipment specially designed for operating in ultra-broad bandwidth (10 GHz) over a wireless link in 140 GHz center frequency. The physical setup of the testbed is shown in the previous chapter, Figure 5.6, and detailed description of the testbed is presented in section 5.4.2.

6.4.3 Neural Network Implementation

The multistage neural network models are implemented in Pytorch [51] platform in the GPU cluster at University at Albany.

RCE Block: The network is trained for 500 epochs with 45,520 training samples and 10,000 validation samples with an SNR range of 5 – 25 dB. This is enough data for training as RCE block operates on LS estimate, which is performed on BPSK modulated constant preamble symbols. We set the regularization term, \( \lambda = 10^{-2} \). Learning rate is set at 5 \times 10^{-3} for each epoch.

EE Block: In this block, the network needs to be trained separately for different modulation orders as we are operating on data symbols. For each modulation order, the network is trained with 12,000 data symbols with SNR range of 5 – 25 dB and 1,880 validation symbols for 700 epochs. Learning rate is set at 2 \times 10^{-4} each epoch. We selected a smaller learning rate, as we have less information about the characteristics of the hardware induced distortions. This extra caution helps the gradient to propagate slowly to avoid any sub-optimal results.
6.5 Complexity Analysis

In this section we will analyze the computational complexity of the proposed network. Given the proposed networks will be trained offline, we consider the complexity of forward pass of the network and discard the back-propagation for complexity analysis.

6.5.1 RCE block

The RCE block primarily consists of fully connected feed forward layers in both encoder and the decoder block as shown in Figure 6.1. So, we will analyze one of them and infer the other from there. In a generic fully connected neural network, let’s denote number of neurons in \( p^{th} \) layer as \( n_p \). To obtain the output of \( p^{th} \) layer, we have to compute a matrix multiplication and subsequent non-linear activation. The shape of the input matrix of \( p^{th} \) layer is \( (N.n_{p-1}) \) and the weight matrix of \( p^{th} \) layer has the shape of \( (n_{p-1}.n_p) \), where \( N \) is the number of input datapoints. Number of flops required to compute the output of \( p^{th} \) layer is \( (N.(2n_p - 1)n_p) \). Additionally, for the activation function of \( p^{th} \) layer, \( (N.n_p) \) flops are required. Total number of flops \( (n_{flops}) \) required for such a network with \( l \) layers is:

\[
n_{flops} = \sum_{p=1}^{l} (N.(2n_{p-1} - 1)n_p) + (N.n_p) \\
\approx \sum_{p=1}^{l} (N.(2n - 1)n) + (N.n) \quad \text{(generalizing)} \\
= \sum_{k=1}^{l} N.2n^2 \\
= 2lNn^2 \quad (6.4)
\]

In the RCE block \( l = 4 \) and in practice \( N = 1 \). Plugging in, the maximum number of flops required is:

\[
n_{flops}^{RCE} = 4.1.(2n^2) \approx 8n^2 \quad (6.5)
\]

Since we are implementing this for higher frequency ultra-broadbands, it is likely to have a larger \( n \) which is dependent on number of subcarriers. On the contrary, we don’t expect
need for additional layers to represent more complex functions, as the goal of RCE block is to mitigate mostly noise and wireless channel distortions. Thus, we can safely declare the complexity of RCE block to be $O(n^2)$.

6.5.2 EE block

To analyze the computation complexity of EE block, we need to take care the unique features of it: 1) input, weights and biases are all complex numbers, 2) each individual complex multiplication requires 4 flops and each addition requires 2 flops, 3) $k$ identical networks are trained/validated in a parallel manner. We are maintaining other general assumptions discussed in RCE block complexity here as well.

To obtain the output of $p^{th}$ layer, we have to compute a matrix multiplication and subsequent non-linear activation. The shape of the input matrix of $p^{th}$ layer is $(N.n_{p-1})$ and the weight matrix of $p^{th}$ layer has the shape of $(n_{p-1}.n_p)$, where $N$ is the number of input datapoints. To compute the output of $p^{th}$ layer, we need $(N.n_{p-1}.n_p)$ complex multiplications and $(N.(n_{p-1} - 1).n_p)$ additions. The activation function of $p^{th}$ layer requires $2(N.n_p)$ flops ($(N.n_p)$ flops for real and imaginary components each) leading to a total number of flops for such network (taking in consideration each complex multiplication is worth 4 flops and addition is worth 2 flops) with $l$ layers:

$$n_{\text{flops}} = \sum_{p=1}^{l} 4(N/k.n_{p-1}.n_p) + 2(N/k.(n_{p-1} - 1).n_p) + 2(N.n_p)$$

$$\approx \sum_{p=1}^{l} 6(N/k.n^2) - (N/k.n) \quad \text{(generalizing)}$$

$$\approx \sum_{p=1}^{l} 6N/kn^2$$

$$= 6ln^2$$  \hspace{1cm} (6.6)

We can plugin $N = 1$ to obtain $n_{\text{flops}} = 6ln^2$. EE block may require a higher number of layers to get rid of hardware induced non-linearities. But that will eventually increase $k$ as well compensating each other, since, in all likelihood, increasing non-linearity is caused
by higher frequency selectivity irrespective of its source. Thus, without loss of generality, we can simply state complexity of EE block is $O(6n^2)$ or $O(n^2)$.

6.6 Evaluation

6.6.1 Real vs. Complex NN

Our proposed architecture is guided by certain hypotheses as discussed in section 6.2.2. The first reasoning is that complex NN will be able to explore the correlation between real and imaginary components. Hence, we implemented a real network similar to the EE block with I and Q concatenated at the input. Figure 6.5a shows the comparison of training loss in real vs imaginary network of one of the 12 parallel units with QPSK modulated data, keeping other conditions and hyperparameters unaltered. We notice that the real network converges to a sub-optimal loss function value with respect to the complex network.

6.6.2 Dataset Requirement

Our second conjecture was that a network requires $M^{Nd}$ set of data to be trained to observe the feature space. To prove our conjecture, we trained a NN with BPSK modulated OFDM symbols with 64-pt FFT and only AWGN noise. The training process has a restricted view to the unique OFDM symbols and only sees a handful of them. This training dataset of 13,880 different OFDM symbols (out of close to $2 \times 10^4$ unique symbols) is chosen to be same as we have in our performance evaluation. Although the training loss reduces significantly, the testing loss does not decrease, leading to high BER even at high SNRs as...
shown in figure 6.5b. This proves that training with a feasible set of data is a true limitation and not caused by any other possible sources of error.

6.6.3 Other NN Architectures

We have compared BER of QPSK modulated symbols of other possible NN implementations with the proposed architecture, as shown in figure 6.5c. Our proposed model outperforms all other solutions, including LS estimate based decoder. We implemented a complex NN similar to EE block, without separating them into parallel units and trained with same amount of symbols as ours. Because of a limited view of the complete dataset during training process, the model is unable to extract all relevant features, which is reflected in the BER (labeled as EE-no splitting). We also implemented a NN model where each subcarrier is treated as individual input for the equalizer network as done in 70 (labeled as individual SC input). The BER improvement here is limited and does not reach the BER of the proposed model as the network does not explore correlation of samples within the coherence bandwidth, and every subcarrier is treated independently.

6.6.4 Performance of Proposed Model

Figure 6.6 shows the BER variation for an SNR range 5-25 dB for four different modulation orders compared to that of a) LS based traditional receiver and b) approximate LMMSE (ALMMSE) 58 based receiver. We observe that the proposed model achieves improved BER for all modulation orders over both LS and ALMMSE based receiver - upto 6 dB over LS based receiver and 4 dB over ALMMSE receiver in BPSK, 4.4 dB over LS receiver and 2.1 dB over ALMMSE receiver in QPSK, upto 2.2 dB over LS receiver and 1.4 dB over ALMMSE receiver in QAM 16 and 1.2 dB over LS receiver and 0.8 dB over ALMMSE receiver in QAM 64 modulated data. From visual inspection, it is hard to verify consistency in improvement in BER for all modulation orders. To compensate that, we provided a reference line at a bit error rate $3.5 \times 10^{-2}$. Around this reference line for all four different modulation orders, we observe improvement of 3.2-4 dB over LS based receivers in practice.
6.7 Conclusion

This work advances the effectiveness of using deep learning in improving OFDM physical layer for THz band communication. This is achieved by injecting domain knowledge to modify the neural network architecture and eventually design a complete OFDM receiver. It also paves the way to develop OFDM systems for emerging spectra, like THz, that are powered by AI to overcome the shortcomings of practical communication systems.
CHAPTER 7

Conclusion

In this thesis, we have developed and rigorously analyzed deep learning models developed with an anchor to the theoretical background of communication systems for both existing sub-6 GHz systems and emerging systems in the THz spectrum. The proposed design addresses specific major issues in the respective spectra. This degree of subjectivity and specificity is desirable instead of blatant generalization which may fail to address the diverse issues of different bands of the RF spectra.

From our multiple simulations and testbed experiment results, we can firmly conclude that empowering NN architecture with domain knowledge greatly improves systems performance and alleviates drawbacks of NN based designs such as enormous data requirement and large computation cost for training purposes as the network becomes more robust and eventually require less training data and time. This work promotes development of interpretable NN systems.

We have successfully shown that the channel estimation setup developed for sub-6 GHz systems significantly improves the estimate over the tools currently in practice even in presence of severe channel. We have developed an effective waveform for THz communication system that is able to tackle the issues caused by ultra-broadband. Additionally, development of the NN based receiver design for THz communication system has provided us the ability to reliably achieve a physical layer datarate as high as 43 Gbps leading towards the overarching goal of satisfying the ever growing demand for reliability and high data rate in communication systems. While we await further real time implementations and tests, this work paves the path to further development of deep learning and AI powered physical layer in current and xG communication systems.


[7] Alexandros-Apostolos A. Boulogeorgos, Evangelos N. Papasotiriou, and Angeliki Alexiou, A distance and bandwidth dependent adaptive modulation scheme for THz commu-


