# Machine Learning for Data-Driven Signal Separation and Interference Mitigation in Radio-Frequency Communication Systems

by

Cheng Feng Gary Lee

B.S., Stanford University (2016)
S.M., Massachusetts Institute of Technology (2019)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2023

Authored by: Cheng Feng Gary Lee
Department of Electrical Engineering and Computer Science
August 30, 2023

Certified by: Gregory W. Wornell
Sumitomo Professor of Engineering
Thesis Supervisor

Accepted by: Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

# Machine Learning for Data-Driven Signal Separation and Interference Mitigation in Radio-Frequency Communication Systems

by

Cheng Feng Gary Lee

Submitted to the Department of Electrical Engineering and Computer Science
on August 30, 2023, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

## Abstract

Single-channel source separation for radio-frequency (RF) systems is a challenging problem relevant to key applications, including wireless communications, radar, and spectrum monitoring. This thesis addresses the challenge by focusing on data-driven approaches for source separation, leveraging datasets of sample realizations when source models are not explicitly provided. To this end, deep learning techniques are employed as function approximators for source separation, with models trained using available data. Two problem abstractions are studied as benchmarks for our proposed deep-learning approaches. Through a simplified problem involving Orthogonal Frequency Division Multiplexing (OFDM), we reveal the limitations of existing deep learning solutions and suggest modifications that account for the signal modality for improved performance. Further, we study the impact of time shifts on the formulation of an optimal estimator for cyclostationary Gaussian time series, serving as a performance lower bound for evaluating data-driven methods. The thesis also introduces the "RFChallenge" as a benchmarking platform, aimed at addressing the gap in current literature for a comprehensive comparison of emerging machine learning solutions for RF signal separation. Finally, we explore an alternative approach of using deep learning to train a library of individual signal models that can be used together for subsequent inference tasks. While showing promise as a scalable strategy for the problem, our preliminary findings uncover the practical limitations of such methods. Ultimately, this thesis seeks to provide insights into judicious choices of data-driven solution architecture based on the signal structures under consideration. Our findings aim to stimulate further research at the intersection of machine learning and RF system design, contributing to the development of next-generation wireless technology through data-driven methodologies.

Thesis Supervisor: Gregory W. Wornell
Title: Sumitomo Professor of Engineering

# Acknowledgments

First and foremost, I would like to extend my heartfelt gratitude to my supervisor, Professor Gregory Wornell. Over the past six years, his patience, support, and mentorship have been invaluable to me. He has generously provided me with the space to explore and learn, while consistently offering timely guidance and advice throughout this journey.

Equally, I would like to thank Professor Yury Polyanskiy, who, over the course of this project, effectively assumed the role of my second supervisor. The wisdom I gleaned from our regular interactions not only directed the trajectory of this project but has also paved the way for my future explorations. His insights have consistently been both enlightening and inspiring.

A special mention goes to Dr. Binoy Kurien. Navigating the practical realities of this application space would have been far more challenging without his willingness to invest his time and energy. Through his dedication, he has instilled in me a profound appreciation for the tangible significance of our work, grounding my academic pursuits in practical reality.

Additionally, I sincerely thank Professor Lizhong Zheng, who generously took the time to be on my thesis committee, and provided his fresh perspective on my work. His keen insights and thoughtful feedback have added clarity to my understanding. The time he has taken to discuss my work has enriched my thinking and helped refine my approach.

I also extend my deepest gratitude to my collaborators on this thesis project: Amir, Alejandro, Yuheng, Jennifer and Tejas. Beyond being exceptional peers and colleagues in this endeavor, they have been profound sources of inspiration. Their guidance and mentorship have been instrumental in my growth, honing my skills and molding me into the researcher I am today. Their influence has undeniably enriched and elevated my academic journey.

I would also like to express my gratitude to the members, both past and present, of the Signals, Information, and Algorithms group. Our regular, insightful interactions have been particularly enriching, enlightening me about the diverse projects being pursued. Additionally, a special acknowledgment is due to Tricia for her unwavering assistance throughout this journey.

To all my friends, both old and new, who have accompanied me on this journey—there are perhaps too many to name. In particular, Angus, Chenyang, Gladia, Lingxiao, Morris, Ning—their steadfast support during my time at MIT, through challenges in our academic

programs and in life, has been my bedrock. I am also immensely grateful to my longtime friend Kenneth, who, despite geographical distances, has remained an intellectual peer in many facets of my life.

To my partner, Clare, her unwavering support, both in navigating the challenges and in celebrating my successes (no matter how small), has been my cornerstone.

Lastly, but most certainly not least, my deepest thanks go to my family. Despite the 9,000 miles distance and the 12-hour time difference that separates us, their unyielding emotional support and love have been ever-present and always felt.

# Contents

**1 Introduction 17**

1.1 Historical Challenge of Source Separation . . . 17

1.2 Deep Learning for Separating Signals . . . 20

1.3 Deep Learning in the RF Domain: A New Frontier? . . . 21

1.4 Zooming In: Single-Channel RF Source Separation . . . 23

1.5 Simple Demonstrations of Signal Separation . . . 24

1.5.1 Separating i.i.d. Gaussians . . . 24

1.5.2 Separating Multivariate Colored Gaussians . . . 25

1.5.3 Separating i.i.d. BPSK Symbols . . . 26

1.6 Structure of the Thesis . . . 28

**2 Preliminaries 31**

2.1 Properties of Discrete Time Signals . . . 31

2.2 Properties of Digital Communication Signals . . . 33

2.2.1 Digital Modulation Systems . . . 34

2.2.2 Finite Symbol Set of Discrete Magnitudes . . . 36

2.3 Matched Filtering . . . 36

2.4 Signal Estimation . . . 39

**3 On Neural Architectures for Separating OFDM Waveforms 43**

3.1 Motivation: Learning to Separate OFDM Waveforms . . . 45

3.2 Related Works on Deep Learning with OFDM Waveforms . . . 46

3.3 Problem Formulation . . . 47

3.4 Model-Based Insights to the Problem . . . 49

3.5 End-to-end Separator via Deep Neural Networks . . . 51

3.6 Computational Simulations . . . 52

3.7 Insights on OFDM Domain-informed Architecture . . . 55

3.7.1 Proposed Neural Architecture Modifications . . . 55

3.7.2 Features of Long 1st-Layer Convolutional Kernels . . . 56

3.8 Concluding Remarks . . . 58

**4 Benchmarking with Cyclostationary Gaussian Signals 61**

4.1 Background on Cyclostaionary Signals . . . 62

4.1.1 Cyclostationarity . . . 62

4.1.2 Stationarizing and Ergodicity . . . 63

4.2 Problem Formulation . . . 65

4.3 Optimal Model-Based Estimators . . . 66

4.3.1 Case 1: The Optimal Linear MMSE Solution . . . 67

4.3.2 Case 2: The Oracle MMSE Solution . . . 68

4.3.3 Case 3: The Optimal MMSE Estimator . . . 69

4.4 Methodologies for Data-Driven Signal Separation . . . 69

4.4.1 Difficulties in Realizing the MMSE Estimator . . . 70

4.4.2 Supervised Separation with U-Net . . . 70

4.5 Computational Simulation . . . 72

4.5.1 Signals from Randomly Generated Covariances . . . 72

4.5.2 Communication-like Waveforms . . . 74

4.6 Residual Error Statistics . . . 76

4.7 Expected MSE for representative cases . . . 76

4.8 Going beyond Gaussianity . . . 80

**5 Parameters and Hyperparameters for Deep Learning Architectures 83**

5.1 Recap: Specifications of Waveforms under Study . . . 83

5.1.1 Implementation Details . . . 84

5.1.2 Hyperparameters for Training . . . 85

5.2 End-to-End Separation versus Demodulation Approaches . . . 87

5.3 A Need for Benchmarks . . . 90

**6 Computational Experiments with Real-World RF Signals 91**

6.1 RFChallenge with RF Signal Recordings . . . . . . . . . . . . . . . . . . . . . 92

6.2 Implementation Details . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 95

6.3 Baseline Results for Demodulation Sub-Challenge . . . . . . . . . . . . . . . . 96

6.4 Baseline Results for Separation Sub-Challenge . . . . . . . . . . . . . . . . . . 99

6.5 Technical Challenges . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 100

**7 Towards Scalability with Library of Learned Priors 103**

7.1 Information-Theoretic MMSE . . . . . . . . . . . . . . . . . . . . . . . . . . . 104

7.2 MAP Estimation Approach to Single-Channel Source Separation . . . . . . . 105

7.2.1 Case Study with Multivariate Gaussian . . . . . . . . . . . . . . . . . 106

7.2.2 Case Study with Learned Models . . . . . . . . . . . . . . . . . . . . . 110

7.3 Discussion and Concluding Remarks . . . . . . . . . . . . . . . . . . . . . . . 114

**8 Conclusion 117**

8.1 Challenges and Opportunities . . . . . . . . . . . . . . . . . . . . . . . . . . . 118

8.2 Bridging Model-Based and Data-Driven Approaches . . . . . . . . . . . . . . . 119

8.3 Concluding Remarks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 120

# List of Figures

1-1 Covariance Structures for illustrative example in Subsection 1.5.2 . . . . . . . 25

1-2 Comparison of Time-Averaged MSE when one uses the true multivariate Gaussian statistics versus the Gaussian statistics from assuming temporally i.i.d. statistics. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26

2-1 Block diagram of a typical communication pipeline. The interference mitigation block (red) refers to a potential pre-processing stage of the received signal, by separating the interference from the signal-of-interest, before the demodulation and decoding of the received signal—this interference mitigation/source separation block is the primary focus of this thesis. . . . . . . . . 33

2-2 Visualizations of key symbol constellations primarily used in this thesis. . . . 37

2-3 Block diagram of the matched filtering demodulation pipeline. . . . . . . . . . 38

3-1 Selected neural architectures used in audio separation. . . . . . . . . . . . . . 44

3-2 Visualization of OFDM structure. . . . . . . . . . . . . . . . . . . . . . . . . 50

3-3 Features of a selected subset of the kernel weights on the first convolutional layer. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 57

4-1 The U-Net architecture used for source separation in our simulations. . . . . . 71

4-2 Visualization of (a sub-matrix of) the covariance matrices for the signals used in the respective examples. . . . . . . . . . . . . . . . . . . . . . . . . . . . 73

4-3 Separation performance of the U-Net separator vs. optimal model-based estimators for waveforms with randomly generated covariance structures. . . . . 74

4-4 Separation performance of the U-Net separator vs. optimal model-based estimators for communication-like waveforms. . . . . . . . . . . . . . . . . . . . . 75

4-5 MSE across the time indices . . . . . . . . . . . . . . . . . . . . . . . . . . 77

4-6 Expected MSE for different signal lengths, plotting for configuration at $-30$ dB SINR and for $\tau_s = 0$ and $\tau_b = 40$. . . . . . . . . . . . . . . . . . . . . . . . . 78

4-7 Expected MMSE for different configurations . . . . . . . . . . . . . . . . . . . 79

4-8 Expected MSE for when interference contains an AWGN component which is magnified by the gain $\kappa$. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 80

4-9 Comparison of MSE and BER in the single-channel source separation of digital communication signals. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 82

5-1 Variability in the training and validation losses between independent runs when training the Separation U-Net. . . . . . . . . . . . . . . . . . . . . . . . 85

5-2 The effect of different training set sizes on the training and validation losses when training the Separation U-Net. . . . . . . . . . . . . . . . . . . . . . . . 86

5-3 The effect of different mini-batch sizes on the training and validation losses when training the Separation U-Net. . . . . . . . . . . . . . . . . . . . . . . . 87

5-4 The effect of different first-layer kernel sizes on the training and validation losses when training the Separation U-Net. . . . . . . . . . . . . . . . . . . . . . 88

5-5 The effect of first-layer kernel size on the test performance, in terms of MSE and BER, across different SIR levels. . . . . . . . . . . . . . . . . . . . . . . . 88

5-6 Comparison of Neural Network Methods—End-to-End Separation versus Demodulation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 89

6-1 Representative frames of the dataset provided in the RFChallenge. . . . . . . 93

6-2 Representative visualizations of the SOI under considerations for the Demodulation Sub-Challenge. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 94

6-3 Representative visualizations of the signal mixtures in the RFChallenge Single-Channel Source Separation . . . . . . . . . . . . . . . . . . . . . . . . . . . . 95

6-4 Comparison of MSE in the signal extraction and reconstruction for the single-carrier communication SOI for the Demodulation Sub-Challenge. . . . . . . . 97

6-5 Comparison of BER in the interference mitigation for the single-carrier communication SOI for the Demodulation Sub-Challenge. . . . . . . . . . . . . . 98

6-6 Comparison of MSE in the signal extraction and reconstruction of the Comm-Signal2 SOI component (whose signal model is not explicitly provided) for the Separation Sub-Challenge. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 99

6-7 Comparison of the CRC Success Rate when a standard demodulation tool (specific to CommSignal2) is applied to the extracted signal component in the Separation Sub-Challenge. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 100

7-1 Comparing the MSE and BER of the estimated SOI from Algorithm 1 against traditional linear processing methods and against using individual MMSE denoisers only. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 113

# List of Tables

3.1 MSE (in dB) of the extracted SOI using audio-domain neural networks. . . . 53

3.2 MSE (in dB) of the extracted SOI using our proposed architecture, the modified Wave-U-Net. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55

3.3 MSE (in dB) of the extracted SOI using the modified Wave-U-Net with different first-layer kernel sizes. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 56

# Chapter 1

# Introduction

The proliferation of wireless radio-frequency (RF) devices today is leading to an overly crowded radio environment and a growing scarcity of spectrum resources. Consequently, spectrum sharing has become unavoidable, and different wireless systems may coexist within the same frequency bands. For instance, the 2.4 GHz ISM band[1], utilized by standard wireless communication systems like 802.11 WiFi and Bluetooth, is susceptible to interference from multiple sources. This includes cross-technology interference, where WiFi and Bluetooth signals can interfere with each other, as well as unintended interference from common household appliances such as microwave ovens. The presence of such interference poses significant challenges for communication systems operating within this frequency band.

In this context, the task of separating different RF signals from a mixed recording becomes crucial for further processing, analysis, and characterization [1]. Such a capability would be particularly helpful for RF scene analysis, where the identification of various RF devices operating in a particular part of the spectrum is of interest. Furthermore, the accurate separation of a signal-of-interest from interference is essential to ensure the quality and reliability of communication systems within this crowded radio environment [2].

## 1.1 Historical Challenge of Source Separation

The broader interest in signal separation extends beyond the RF domain and is motivated by the need to extract valuable information from a complex and often noisy world. Such a problem has been of long-standing interest, with its roots tracing back to as early as the

[1] ISM bands refer to frequency bands designated for industrial, scientific and medical (ISM) applications.

mid-19th century, when researchers marveled at humans' ability to distinguish individual audio sources from a cacophony of sounds. As described by Helmholtz back in 1863,

> ...in the interior of a ball-room, for instance. Here we have a number of musical instruments in action, speaking men and women, rustling garments, gliding feet, clinking glasses, and so on. All these causes give rise to systems of waves... in short, a tumbled entanglement of the most different kinds of motion, complicated beyond conception. And yet... the ear is able to distinguish all the separate constituent parts of this confused whole... [3, p. 26-27].

This is indeed one of the longstanding challenges in artificial intelligence (AI)—i.e., replicating the ingenuity of human perception in non-biological systems. Nearly a century later, Cherry referred to this as the "cocktail party problem" [4], and mentioned in his book that "no machine has yet been constructed to do just this" [5, p. 278]. We have come a long way in the past 150 years, with AI systems now performing well and perhaps even attaining superhuman performance in some instances [6, 7].

The cocktail party problem, also known as the ***source separation*** problem, has been well-studied over the past century with many essential applications beyond the audio domain. Unsurprisingly, this problem is also highly relevant to wireless RF systems [8–10]. In a "cocktail party" of wireless devices, a transmitter-receiver pair might seek to focus on their communication chain and mitigate interference effects within the shared space, both physically and spectrally.

Source separation is a well-studied problem with many important applications in RF systems and wireless communication [8–10], among many others.

One typical formulation is the blind source separation problem, whereby the details about the constituent source signals and the mixing system are unknown, and the signals are separated based on the observed mixture alone—i.e., "blindly". Nevertheless, implicit assumptions and conditions are typically introduced to such problems. A popular framework to tackle the blind source separation problem is independent component analysis (ICA) [11, 12]. This approach leverages the spatial diversity available in measurements from multi-antenna receivers, as well as the underlying assumption about statistical independence in the signal sources present.

On the other hand, a particularly challenging problem of interest in this space is the *single-channel* (or single-sensor) source separation, where the RF receiver is more constrained, which presents a different set of challenges. Such a setting has recently gained significant interest [7, 13–15]. In this regime, the aforementioned algorithms are irrelevant due to the absence of spatial diversity. Instead, we have to exploit the temporal structures of the latent sources to achieve good separability. The focus of this thesis is on the single-channel regime of this problem.

Several non-ICA methods have been proposed for single-channel source separation in digital communication, including maximum likelihood sequence estimation of the target signal. In practice, these methodologies focus on a model-based approach to extract the underlying information bits/symbols of the communication signals, employing algorithms such as particle filtering [16] and per-surviving processing algorithms [17]. However, these algorithms operate under the assumption of known source models for both the SOI and the interference signal (e.g., separation of two QPSK signals [17], which is more applicable to a multiple access channel setting). In this case, the key idea is to determine what set of underlying information bits agrees best with the observation; and while the combinatorial search is impractical, the algorithms proposed offer computationally tractability to such inference procedures. Even so, the complexity of such algorithms scales exponentially with modulation order [1], limiting its applicability on modern communication waveforms that tend to adopt higher-order constellations. Furthermore, these methods are not applicable in uncoordinated settings, where there is no coordination or alignment between the signal-of-interest and the co-channel interference, and that the interference model is generally unknown.

Some strategies exploit the inherent properties of conventional wireless devices. For instance, conventional RF devices have been designed to operate in an orthogonal fashion—e.g., operating in different bands on the frequency spectrum or at different times (not to transmit when other devices are occupying the spectrum). These measures prevent spectral or temporal overlap of source components, enabling the mitigation of interference via linear filters or time-frequency masking.

For such a setting, if the sources are separable in time and/or frequency, one could separate them via masking in the spectrogram or classical filtering methods, e.g., [18, 19]. The primary challenge lies in separating co-channel signals, where the sources overlap (partially/fully) in both time and frequency, necessitating the development of novel interference mitigation approaches.

Furthermore, in practical scenarios, prior knowledge of the signal models may not be known or readily available. Perhaps a more realistic approach is to assume that only a collection of the underlying communication signals ("dataset") is available. This can be obtained, for example, through direct or background recordings, or using high-fidelity simulators (e.g., [20, 21]), allowing for a *data-driven* approach. For example, the single-channel "RFChallenge" [22] focuses on the data-driven single-channel source separation problem, providing raw datasets of various RF signals provided with minimal to no information about their generation processes.

This work focuses on the data-driven flavor of the signal separation problem, particularly in the context of co-channel RF signals. Without providing source models, the research explores how to approach such problems. And while general model-based approaches can be adopted by making simplifying assumptions that lead to mathematically tractable models, the emphasis is on leveraging data to exploit stronger structures present.

## 1.2 Deep Learning for Separating Signals

With interest in cases where the source components' model is unknown, we look to the potential of data-driven methods in such scenarios. Notably, as strategies to tackle this problem evolve, new methodologies and tools have emerged, especially in the current age of AI and machine learning. This is particularly prevalent in the audio and image domains, where the availability of vast datasets and benchmarks has fueled progress. Given the rapid advancements made with deep learning in these areas, we aim to assess whether these techniques can be harnessed effectively for the long-standing source separation problem, particularly in the single-channel case.

Recent efforts demonstrate the successes of deep neural network techniques for source separation in the single-channel regime for image and audio counterparts [23–26]. These methods typically exploit the inherent structure specific to the signal type. For example, natural images may be separable by color features and local dependencies [25], whereas speech signals are commonly addressed by time-frequency spectrogram masking [27–29]. Furthermore, for time series (1-dimensional) data, if the sources are separable in time and/or frequency, appropriate spectrogram-based masking and classical filtering methods can also be adopted, e.g., [18, 19].

Considering the source separation of time series, there have also been works, such as those involving audio signals, where methods perform signal separation in the time domain [23, 30–34] (which we will look into with more detail in Chapter 3), in contrast to spectrogram-based methods. The assumption that similar neural architectures used in such time-domain audio-based source separation would naturally extend to RF signal separation may seem plausible due to the time-series nature of both types of signals. Nevertheless, it is also critical to consider the inherent differences and unique challenges presented by RF signal separation. Methods designed for speech and music processing may not necessarily be transferable to other types of time series data, such as RF signals, due to their fundamentally different characteristics. On the one hand, recent work demonstrated the effectiveness of one of these architectures, DPRNN, in separating the time-domain representation of seismic signals [35]. On the other hand, many of the aforementioned audio-separation methods hinge upon learning an effective masking operator on the latent representation, which may not necessarily be as effective on co-channel signals.

In the realm of RF signals, research on time-domain separation using neural networks is limited but has seen growth in the last five years. It is worth noting that these works have largely concentrated on single-carrier signals and radar signals [36–39]. To our knowledge, apart from our recent works [40–42], we are not aware of any other model-blind signal separation approaches involving mixtures with other RF signals, particularly multicarrier OFDM waveforms. At the same time, we acknowledge that the comparative evaluation of these methodologies poses a considerable challenge due to the lack of a unifying benchmark, making direct comparisons difficult. We identify this as a gap in the current literature and aim to make contributions in this direction, as is discussed later in this thesis.

## 1.3 Deep Learning in the RF Domain: A New Frontier?

Deep learning has brought about transformative changes in several fields, notably to image, audio, and natural language processing. However, its potential impact on RF systems remains a topic of ongoing exploration. The uncertainty arises as we question whether deep learning can offer the same benefits to RF signals as it does in other domains. Many conventional signal processing algorithms for RF communication and sensing systems, which are based on statistical models, are typically designed to be provably optimal for mathematically tractable signal models (e.g., linearity, stationarity, Gaussianity). Yet, these methods can still perform reasonably well in practice as long as these models sufficiently capture real-world effects. This is coupled with decades of prior works that taps into rich domain knowledge and complex engineering expertise. As a result, the gains one might obtain from deep learning methods could be marginal, or even insignificant [43]. Furthermore, deep learning for RF systems is still in its infancy, and it lacks the same level of solid analysis and performance guarantees compared to its model-based counterpart.

However, emerging applications with wireless technologies are giving rise to more complex scenarios, which may not be adequately modeled by the mathematically simple models that were alluded to earlier. Traditional human-in-the-loop processes for statistical modeling and expert-designed solutions may not scale well in these situations; on the other hand, deep learning methods offer the potential to capture the underlying complex structures from data with minimal engineering by hand. In the past few years, we have seen a growing body of work that considers how deep learning can address these more complex scenarios in a selected number of problems, such as signal identification [44, 45], modulation classification [21, 43], and waveform/codebook optimization [46–48], to name a few. We believe that this body of work will lead to new insights into how AI can benefit RF systems, and pave the way for a paradigm shift in the design and development of next-generation wireless technologies. In this thesis, we study the problem of data-driven source separation and signal estimation, which are essential and relevant to building RF systems with enhanced spectral awareness and interference rejection capabilities. Further, we propose using machine learning, particularly deep learning tools, to advance the state-of-the-art in this field.

A recurring setting in these problems is that the complete source models are not given, but instead, we have access to datasets of sample realizations, thereby motivating data-driven approaches. We compare the performance of these methods against conventional model-based approaches[2] to evaluate how well they perform and capture the underlying model from data. This investigation highlights the need for appropriately designed tools that bridge the gap between data-driven machine-learning methods and optimal model-based approaches.

[2] with source models provided through a genie, thereby serving as a performance bound/baseline

## 1.4 Zooming In: Single-Channel RF Source Separation

With this foundation, we delve deeper into the mathematical formulation of our problem. This thesis primarily focuses on single-channel source separation, specifically concerning a two-component mixture. This defines the scope of signal separation explored herein. In such a setting, we can also view it as a signal estimation or target signal extraction, since estimating one component naturally gives rise to the recovery of the other (by subtracting the former from the original mixture). For the purposes of this thesis, we broadly refer to these as source separation; however, we note that the general form involving more sources may necessitate a different approach and potentially a more complex formulation.

We now formalize the signal separation problem. Consider the following model of an observed signal of length $N$, which is a noisy mixture of two latent sources,

$$\boldsymbol{y} = \underbrace{\boldsymbol{s}}_{\text{signal-of-interest}} + \underbrace{\boldsymbol{b}}_{\text{interference and noise}}, \tag{1.1}$$

where $\boldsymbol{s}, \boldsymbol{b} \in \mathbb{C}^N$ are the (unobservable) statistically independent signals. For our discussion, $\boldsymbol{s}$ is termed the "reference" signal or the signal-of-interest (SOI), and $\boldsymbol{b}$ is the contributions from interference and noise (broadly viewed as the "signal-not-of-interest").

We also introduce the notion of "signal-to-interference ratio" (SIR)[3], which is effectively the ratio of the average power of the SOI to that of the interference term, i.e., the quantity $\|\boldsymbol{s}\|_2^2/\|\boldsymbol{b}\|_2^2$.

The goal in this signal separation problem is to produce an estimate $\widehat{\boldsymbol{s}}$ based on $\boldsymbol{y}$ so that given some metric $d$, the value $\mathbb{E}\left[d(\widehat{\boldsymbol{s}}, \boldsymbol{s})\right]$ is minimized. We focus our attention on the time-averaged squared error, $d(\widehat{\boldsymbol{s}}, \boldsymbol{s}) = \frac{1}{N}\|\widehat{\boldsymbol{s}} - \boldsymbol{s}\|_2^2$, leading to the time-averaged minimum mean square error (MMSE) criterion.

We are particularly interested in the case where we do not have explicit knowledge of the signal models—i.e., the distributions of $\boldsymbol{s}$ and $\boldsymbol{b}$ are unknown. Nevertheless, we assume we have a dataset of independent, identically distributed (i.i.d.) copies of $\{(\boldsymbol{y}^{(i)}, \boldsymbol{s}^{(i)})\}_{i=1}^{M}$, enabling a data-driven approach.

At different parts of this thesis, we will consider certain slight variations of (1.1)—e.g., noisy versus noiseless cases, explicit modeling of relative gains between the two source terms,

[3] A related term that is also adopted in this thesis is the "signal-to-interference-plus-noise ratio" (SINR), in the event that we consider the summed contributions of the interference and noise in this quantity.

introducing notions of time shifts, or making further assumptions about $\boldsymbol{s}$ and $\boldsymbol{b}$. We will revise the formulation in the appropriate sections of this thesis.

## 1.5 Simple Demonstrations of Signal Separation

In this section, we put forth three simple illustrative examples to demonstrate the viability of single-channel source separation, should the source models be known. This helps set the stage for appreciating when perfect separation is attainable, studying examples when perfect separation is not attainable, and what separation performance one could expect in those scenarios.

### 1.5.1 Separating i.i.d. Gaussians

Suppose that each element of the source vectors are i.i.d. zero-mean real-valued Gaussian random variables, $s[n] \sim \mathcal{N}(0, \sigma_s^2)$ and $b[n] \sim \mathcal{N}(0, \sigma_b^2)$, and that the two sources are statistically independent.

Since the signals are i.i.d. in time, it suffices to look at a single time step, as the statistical characterization applies equally across all time steps.

We observe $y[n]$, and seek an estimate $\widehat{s}$ that minimizes the MSE. Since $s[n]$ and $b[n]$ are jointly Gaussian, the MMSE estimator given $y[n]$ is hence

$$\widehat{s}[n] = \frac{1}{1 + \sigma_b^2/\sigma_s^2}\, y[n] \tag{1.2}$$

and the expected error for each element is

$$\mathbb{E}\left[\|s[n] - \widehat{s}[n]\|_2^2\right] = \left(1 - \frac{1}{1 + \sigma_b^2/\sigma_s^2}\right) \sigma_s^2 = \left(\frac{1}{1 + \sigma_s^2/\sigma_b^2}\right) \sigma_s^2. \tag{1.3}$$

Therefore, the time-averaged MSE is

$$\mathbb{E}\left[\frac{1}{N}\|\boldsymbol{s} - \widehat{\boldsymbol{s}}\|_2^2\right] = \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{E}\left[\|s[n] - \widehat{s}[n]\|_2^2\right] = \left(\frac{1}{1 + \sigma_s^2/\sigma_b^2}\right) \sigma_s^2 \tag{1.4}$$

which is the same as in (1.3) since each entry is i.i.d. in time.

To provide some numerical perspective on the relative magnitudes of the resulting MSE—

- When $s$ and $b$ are of the same power, i.e., SIR of 0 dB, the time-averaged MSE is $1/2\, \sigma_s^2$

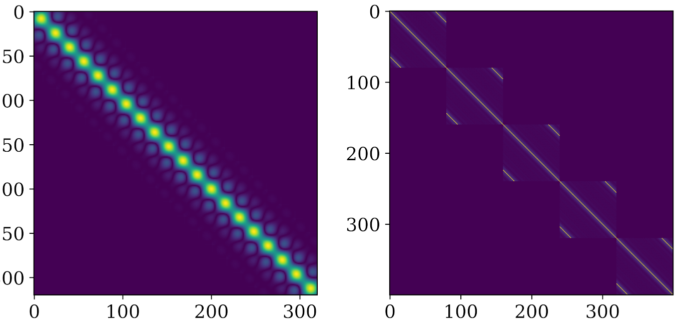


Figure 1-1: Covariance Structures for illustrative example in Subsection 1.5.2

(for a unit power $s$, this corresponds to $-3$ dB MSE).

- When the power of $s$ is half of $b$, i.e., SIR of $-3$ dB, the time-averaged MSE is ${}^{2}\!/_{3}\,\sigma_s^2$ (or around $-1.76$ dB MSE for a unit power $s$).

### 1.5.2 Separating Multivariate Colored Gaussians

Now we consider a vector source, where the elements in time are no longer i.i.d., and therefore the temporal correlations can be exploited for better signal separation performance.

We consider a specific example where $s, b \in \mathbb{R}^N$ ($N$-length real-valued vectors), and they are multivariate Gaussian, $s \sim \mathcal{N}(0, \Sigma_s)$ and $b \sim \mathcal{N}(0, \Sigma_b)$.

Since $s$ and $b$ are jointly Gaussian, the MMSE estimator in this case is a linear one, and can be written as

$$\widehat{s} = \Sigma_s(\Sigma_s + \Sigma_b)^{-1} y \tag{1.5}$$

and the time-averaged MMSE can be expressed as

$$\text{TA-MSME} = \frac{1}{T}\mathbb{E}\left[\|s - \widehat{s}\|_2^2\right] = \frac{1}{T}\text{Tr}\{\text{C}_\text{e}\} \tag{1.6}$$

$$C_e \triangleq \Sigma_s - \Sigma_s(\Sigma_s + \Sigma_b)^{-1}\Sigma_s. \tag{1.7}$$

At the extreme ends, when the SIR tends towards infinity, the time-averaged MSE approaches 0; whereas as the SIR tends to 0 (or negative infinity in decibels), the time-averaged MSE converges to the average power of $s$, i.e., $\frac{1}{N}\text{Tr}(\Sigma_\text{s})$.

It is important to note that the MSE is influenced by the temporal structures of both signal components. For illustrative purposes, we adopt the covariance structures as visualized

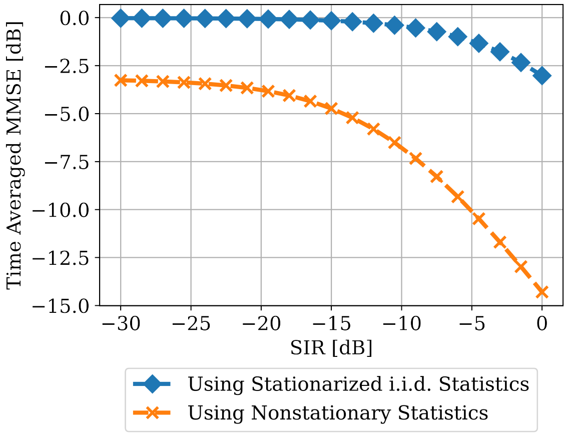


Figure 1-2: Comparison of Time-Averaged MSE when one uses the true multivariate Gaussian statistics versus the Gaussian statistics from assuming temporally i.i.d. statistics.

in Fig. 1-1 [4], and compute the corresponding time-averaged MMSE.

Additionally, we contrast time-averaged MSE obtained by (1.7) against the MSE if one had made the temporally i.i.d. (stationary) assumption about $s$ and $b$, and leading to (1.4). For the latter, the (marginal) variance is computed by

$$\sigma_s^2 = \frac{1}{N}\mathrm{Tr}(\Sigma_\mathrm{s}) \ ; \ \sigma_\mathrm{b}^2 = \frac{1}{\mathrm{N}}\mathrm{Tr}(\Sigma_\mathrm{b}).$$

This comparison is illustrated in Fig. 1-2, where we plot the TA-MSE versus different SIR levels.

Indeed, the performance of the signal separation is dependent on the temporal structures of both signal components—specifically, the temporal correlation in this Gaussian example. The crux lies in the ability to reliably capture the informative statistical structure, and leverage them for improved signal separation.

### 1.5.3 Separating i.i.d. BPSK Symbols

The second example demonstrates how we can exploit specific temporal structures for better signal separation. In particular, we consider models that reflect the digital nature of our waveforms, which departs from the Gaussian model.

Consider the case where $s[n]$ and $b[n]$ are i.i.d. (in time) BPSK symbols, i.e., $s[n], b[n] \in \{-1, +1\}$, taking each value with equal probabilities. Again, it suffices to look at individual

[4] These structures are loosely based upon covariance structures arising from a root-raised cosine pulse shaping function and from a cyclic-prefix repetition structure, respectively.

time steps under the i.i.d. assumption.

We now consider two particular regimes of different SIR, leading to different outcomes. First, consider the case where the observation is

$$y[n] = s[n] + 2\, b[n],$$

namely corresponding to a $-3$ dB SIR. In this case, $y[n]$ takes on four possible values, and each possible value of $y[n]$ uniquely maps to a specific pair of $s[n], b[n]$ values. Hence, upon observing a particular $y[n]$, the corresponding $s[n]$ can be uniquely identified. As such, perfect source separation is attainable, yielding an MMSE of 0.

Next, consider the case where the sources have the same power (SIR 0 dB),

$$y[n] = s[n] + b[n].$$

Here, $y[n]$ only takes on three possible values. Notably, when $y[n] = 0$, there exist two possible $s[n], b[n]$ pairs of solutions which are equally likely. Under such a circumstance, there is a 50% chance of being unable to discern the true values.

The MMSE estimate given $y[n] = 0$ is given by

$$\begin{aligned}\mathbb{E}\left[s[n]|y[n]=0\right] &= -1 \cdot \mathbb{P}\left(s[n]=-1|y[n]=0\right) + 1 \cdot \mathbb{P}\left(s[n]=+1|y[n]=0\right) \\ &= -1 \cdot (0.5) + 1 \cdot (0.5) \\ &= 0,\end{aligned}$$

and the corresponding MMSE at these points is 1. It is interesting to note that the MMSE estimate of $s[n]$ at such points is 0, which is not in the set of true values for $s[n]$. However, this estimate does indeed result in the minimum MSE (as opposed to picking one of the BPSK values at random, which leads to a squared error of 2 on average).

All in all, the MMSE is 0 in some of these cases—50% of the time to be precise—but nonzero in other cases, yielding a squared error of 1 at those time steps. Interestingly, a worse MSE is obtained for the higher SIR configuration here. This reflects an important observation that higher SIR does not automatically mean a strictly better signal estimation. Instead, the performance of signal separation largely hinges upon the joint statistics of the sources.

## 1.6 Structure of the Thesis

This thesis is structured into eight main chapters, each focusing on different aspects of the RF signal separation problem and the application of machine learning and deep neural network methods.

This chapter serves as an introduction, providing a broad motivation for the thesis work and highlighting the relevance of machine learning in the field of source separation for RF systems. It also presents simple examples to demonstrate the viability and limitations that may arise in this signal separation problem. Chapter 2 introduces fundamental concepts that will serve as essential background for the rest of the thesis.

The next part focuses on abstractions of the signal separation problem for the characterization of possible solution structures and comparing proposed methods. Chapter 3 starts with a simplified problem abstraction based on Orthogonal Frequency Divison Multiplexing (OFDM), a modern modulation scheme. This chapter highlights the limitations of existing deep learning-based solutions derived from audio-based single-channel source separation, and proposes relevant model-based modifications to improve performance. Chapter 4 studies the problem abstraction involving cyclostationary Gaussian time series, and how sources of randomness arising from time shifts can impact the form of the optimal estimator. At the same time, the chapter provides a lower bound on the achievable performance, and demonstrates how deep learning methods, with limited explicit knowledge about the source model, can approach this performance lower bound.

Chapter 5 takes a more empirical view of deep learning methods, and focuses on the characterization of hyperparameters and architectural choices for these neural network methods used in this problem. Chapter 6 builds upon the insights and characterizations gleaned in earlier chapters and applies proposed methods to signals representative of waveforms in RF systems. This includes synthetic waveforms that emulate digital communication signals, as well as real-world over-the-air recordings. The chapter introduces a benchmark known as the “RF Challenge”.

Chapter 7 revisits the single-channel source separation with a different approach than discussed in previous sections; rather, we consider the viability of learning denoisers of individual signal types, and how that corresponds to modeling priors for those signals. This chapter then investigates and discusses how these individually trained models can be used together in separating signal mixtures.

Finally, Chapter 8 provides concluding remarks that summarize the results presented in this thesis.

# Chapter 2

# Preliminaries

In this chapter, we begin by presenting an overview of relevant essential concepts and background, which will provide the foundation for the rest of this thesis.

## 2.1 Properties of Discrete Time Signals

Digital signal processing pertains to the analysis and manipulation of discrete-time signals, achieved by sampling a continuous-time signal at discrete, and typically regular, time intervals. This is prevalent in modern-day systems where samples are recorded at set time points.

A discrete-time signal is a sequence of values that can be written as $[x[0], x[1], ...]$ where,

$$x[n] = \tilde{x}(nT)$$

where $\tilde{x}(t)$ corresponds to some continuous-time signal, and $T$ is the sampling period. The reciprocal of the sampling period, $f_s = 1/T$, corresponds to the sampling frequency. The sampling frequency plays an important role in preserving the fidelity of the discrete representation of the continuous-time signal. According to the Nyquist-Shannon sampling theorem, a band-limited continuous-time signal can be perfectly reconstructed from its samples if it is sampled at a frequency that is at least twice the maximum frequency component present in the signal—also known as the Nyquist frequency. In the scope of this work, we consider all signal components, both synthetically generated waveforms and over-the-air recordings, as discrete-time signals. We assume they are sampled either at or above Nyquist frequency, thus precluding the need to account for aliasing in our signal components.

### Discrete-Time Convolution/Cross-Correlation

An important operation fundamental in digital signal processing is convolution. For two discrete-time signals, $x[n]$ and $h[n]$, convolution is defined as

$$x[n] * y[n] \triangleq \sum_{k} x[k] \cdot h[n-k]$$

where $*$ is the discrete-time convolution operator. When $x[n]$ and $h[n]$ have finite support of $n$, padding is commonly introduced to specify the behavior of $x[n]$ and $h[n]$ outside of their supports. This thesis mainly employs zero-padding, where values outside the support are taken to be 0.

A closely related operation is the discrete-time cross-correlation, which is written as

$$x[n] \star y[n] \triangleq \sum_{k} x[k] \cdot h[n+k],$$

where $\star$ is the discrete-time cross-correlation operator. Cross-correlations are equal to convolutions with a time-reversed kernel. Interestingly, in much of deep learning works, the convolutional layer is often applying a cross-correlation rather than a convolution on the input. Nevertheless, they can effectively lead to the same results by flipping the corresponding learned kernel, or if the kernel is symmetric along the time axis.

### Discrete Fourier Transform

Another important tool is the Fourier transform, which is typically used to analyze the frequency content of discrete-time signals. We focus on the Discrete Fourier Transform (DFT), where

$$X[k] \triangleq \mathrm{DFT}\left\{x[n]\right\} = \sum_{k=0}^{K-1} x[n] \exp(j2\pi kn/N).$$

A key parameter in the DFT is the DFT size, $K$, which also corresponds to the frequency bins. The above can be efficiently implemented using the Fast Fourier Transform (FFT) algorithm.

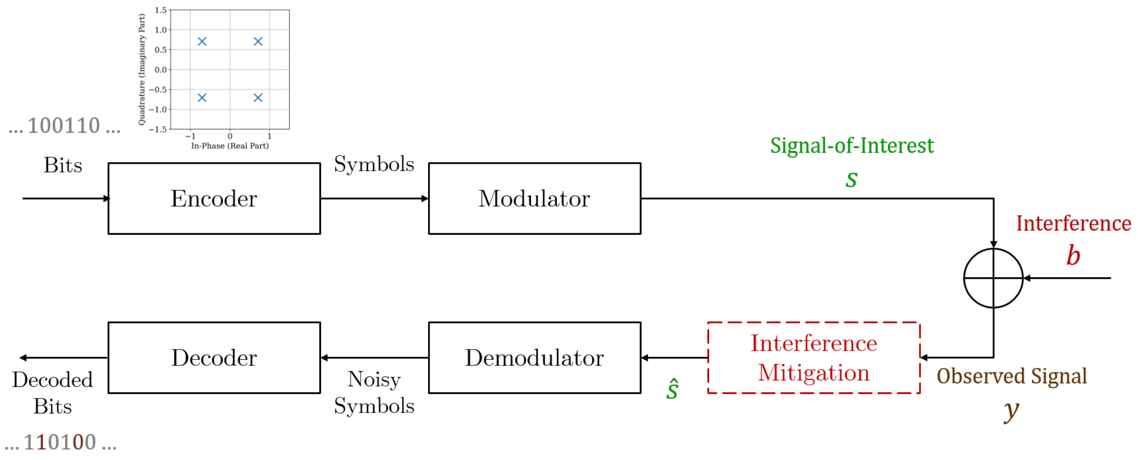


Figure 2-1: Block diagram of a typical communication pipeline. The interference mitigation block (red) refers to a potential pre-processing stage of the received signal, by separating the interference from the signal-of-interest, before the demodulation and decoding of the received signal—this interference mitigation/source separation block is the primary focus of this thesis.

## 2.2 Properties of Digital Communication Signals

In this section, we discuss the properties of some common digital communication signals. These discrete-time signals represent information being transmitted—this thesis focuses on wireless signals in radio-frequency systems. The signals are obtained by modulating a carrier signal with digital data.

For this thesis, we consider the baseband representation, where signals described are independent of the carrier frequency—i.e., demodulated to baseband frequency. Hence, we discuss techniques and representations in baseband processing. The correction and/or estimation of carrier frequencies is not considered within the scope of this thesis.

A typical pipeline is illustrated in Fig. 2-1. For this work, we mainly focus on additive interference and/or additive noise channels. We also include an interference mitigation block (in red), which refers to a possible processing step before demodulation and decoding of the received signal. This step entails separating the interference from the signal-of-interest in our received signal and is the primary focus of this thesis work.

In the following subsections, we further elaborate on the concepts relating to digital modulation and demodulation (particularly, matched filtering), which is subsequently used in our data generation and performance evaluation later in our work.

### 2.2.1 Digital Modulation Systems

Digital modulation is a fundamental technique used in communication systems to transmit digital data over analog channels. It involves mapping digital information to analog waveforms for efficient transmission and reception. Digital modulation systems can be categorized into single-carrier and multi-carrier modulation, each with its advantages and applications.

**Single-carrier modulation systems** transmit data symbols sequentially on a single carrier frequency. One of the most common modulation techniques considered in this thesis is In-Phase/Quadrature (IQ) modulation, which involves mapping bits to two values that are then modulated by the in-phase and quadrature carrier waveforms. The IQ pair can be represented by a single complex number, where the real part represents the in-phase amplitude, and the imaginary part represents the quadrature amplitude. However, communication systems have to operate within a finite bandwidth. In light of this, a root-raised cosine (RRC) pulse-shaping technique is commonly adopted to restrict the signal bandwidth. The choice of RRC filter is particularly effective in minimizing intersymbol interference (ISI) caused by the finite bandwidth of the channel. The desirable properties of RRC make it suitable and widely adopted in many single-carrier systems.

Mathematically, we can represent single-carrier waveforms by

$$s[n] = \sum_{p=-\infty}^{\infty} a_p \, g_p[n - pN_s], \tag{2.1}$$

where $a_p$ are discrete symbols to be transmitted (detailed in the next subsection), $g_p[\cdot]$ is the pulse-shaping filter (e.g., the RRC filter), and $N_s$ is the symbol rate of the signal.

On the other hand, modern systems are interested in more efficient methods to encode information within a finite bandwidth. **Multi-carrier modulation techniques** are developed to meet this demand. This is generally achieved by dividing the available frequency band into multiple subcarriers, each carrying a fraction of the total data. This division allows for parallel transmissions of data symbols, thereby providing better spectral efficiency. A key example is Orthogonal Frequency-Division Multiplexing (OFDM), a multi-carrier modulation technique adopted in various communication standards such as Wi-Fi, 4G LTE, and 5G. OFDM works by dividing the frequency band into multiple closely-spaced orthogonal subcarriers.

The mathematical expression for an OFDM waveform can be written as

$$b[n] = \sum_{p=-\infty}^{\infty} \sum_{\ell=0}^{L-1} a_{p,\ell}\, r[n - p \cdot (L + T_{\text{cp}}) - T_{\text{cp}},\, \ell] \tag{2.2}$$

$$r[n,\, \ell] \triangleq \exp(j2\pi\ell n/L)\, \mathbb{1}_{\{-T_{\text{cp}} \leq n < L\}},$$

where $K$ is the total number of subcarriers, and the coefficients $a_{k,\ell}$ are the discrete symbols to be transmitted. In an OFDM waveform, a cyclic prefix (CP) is typically added before an OFDM symbol, corresponding to a cyclic extension of the symbol. Hence, each OFDM symbol is described for the interval $[-T_{\text{cp}}, K]$, where $T_{\text{cp}}$ refers to the CP length, and $K + T_{\text{cp}}$ corresponds to the total OFDM symbol length (with CP). The finite support of the OFDM symbol is reflected by the finitely supported function $r[n,\, \ell]$.

Note that the expression for individual OFDM symbols without the cyclic prefix (for a single value of $p$) resembles the DFT operation. Indeed, the data symbols can be seen as the coefficients in the frequency domain; therefore, the time-domain waveform is the inverse DFT of these $L$ data symbols/coefficients. Consequently, OFDM is also recognized for its compatibility with efficient receiver designs that leverage the FFT algorithm. It has since become the backbone of many modern wireless communication systems.

Broadly, both single-carrier and multi-carrier signals can be represented as

$$u[n] = \sum_{p=-\infty}^{\infty} \sum_{\ell=0}^{L-1} a_{p,\ell}\; g[n - pT_u, \ell]\; \exp\{j2\pi\ell n/L\}. \tag{2.3}$$

In this context, single-carrier waveforms are special cases where $L = 1$, and OFDM waveforms are particular instances where the filter $g[n, \ell]$ corresponds to a finitely supported rectangular function with a cyclic extension.

The choice of the modulation techniques (i.e., $L = 1$ versus $L > 1$) depends on numerous factors and requirements, depending on the use-case scenarios. Nevertheless, the scope of this thesis does not include choosing parameters and transmission schemes at our discretion. Rather, we study representative examples for both modulation systems in the context of signal separation. Particularly, our objective is to understand the set of challenges associated with each signal type for the problem at hand.

### 2.2.2 Finite Symbol Set of Discrete Magnitudes

In the description of modulation systems earlier, we mention discrete symbols that are being modulated by carrier waveforms. These symbols correspond to a sequence of bits, and are generally represented as complex numbers, which we call the IQ representation—corresponding to the magnitudes that modulate the in-phase and quadrature carrier signals.

Signal constellations are graphical representations of these symbols used in digital communication. These constellations depict the amplitude and phase relationship of all the possible symbols in the corresponding modulation schemes. The three common constellations/modulation schemes under consideration in this thesis are

- Quadrature Amplitude Modulation (QAM): Data is encoded in both phase and amplitude; constellation points are equally spaced in a square grid in the complex IQ plane.

- Phase Shift Keying (PSK): Data is encoded in the phase of the signal, maintaining a constant amplitude; constellation points are equally spaced points along the unit circle in the complex IQ plane.

- Pulse Amplitude Modulation (PAM): We consider this scheme for real-valued constellation points, where only the in-phase (real part) carrier signal is modulated by different amplitudes.

In various parts of the thesis, we also mention Binary PSK (BPSK) and Quadrature PSK (QPSK), which correspond to modulation schemes with two and four possible phases, respectively. Subsequent sections also mention 16-QAM, a square constellation with 16 equally spaced amplitude and phase levels, and 4-PAM, a constellation with 4 equally spaced amplitude levels. These constellations are shown in Fig. 2-2

Note that newer wireless systems are adopting higher-order constellations, which pack more bits into each symbol. Future investigations would benefit from extending the discussions of this thesis to include constellations beyond those mentioned here.

## 2.3 Matched Filtering

Matched filtering is a widely used technique in digital signal processing and communications for detecting and recovering signals corrupted by noise or interference. It exploits knowl-

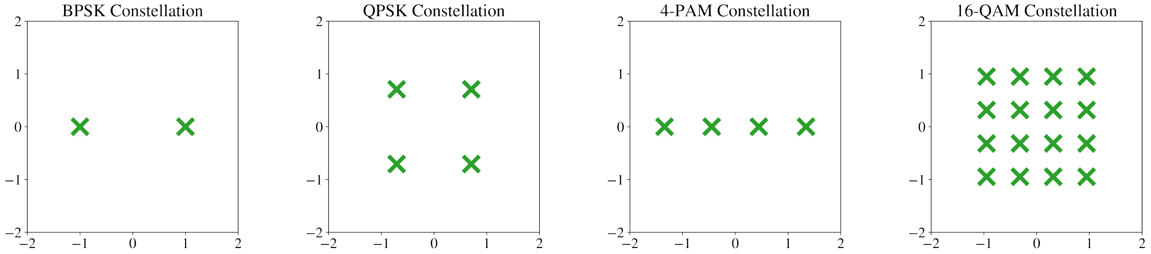


Figure 2-2: Visualizations of key symbol constellations primarily used in this thesis.

edge about the signal waveform to enhance the detection and recovery of the transmitted symbols/bits, and is optimal in the maximum-SNR sense for signals with additive Gaussian noise. While there are various approaches to deriving the matched filter, our focus centers on maximizing the output SNR of our SOI; other perspectives, such as minimizing the probability of detection error under Gaussian noise, are discussed in other works, to which readers are referred to for a more detailed understanding of matched filtering [49].

The basic principle involves filtering (or similarly viewed as cross-correlating, as in Section 2.1) the received RF waveform with a known reference waveform called the "matched filter". The goal is to maximize the signal-to-interference-plus-noise ratio (SINR) at the filtered output, which in turn minimizes the error probability in the subsequent symbol detection step under the Gaussian noise assumption. In the following segment, we develop the theory of the matched filter, following a similar development as presented in [50] but with a specific focus on our problem formulation. Particularly, our discussion centers on matched filtering for single-carrier RRC signals, as this is its primary usage in this thesis.

Consider the baseband RRC-QPSK signal. Suppose, for the purposes of this exposition, that we adopt a simple additive white Gaussian noise (AWGN) channel model, thereby representing our received signal as

$$\begin{aligned} y(t) &= \sum_p a_p \, g_{\text{tx}}(t - pT_s) + w(t) \qquad (2.4) \\ &= g_{\text{tx}}(t) * \sum_p a_p \, \delta(t - pT_s) + w(t), \qquad (2.5) \end{aligned}$$

where $a_p$ are the symbols from a QPSK constellation, $*$ denotes the convolution operator, $\delta(\cdot)$ is the Dirac delta fucnction, and $w(t) \sim \mathcal{N}(0, \sigma^2_{\text{AWGN}})$ is the additive noise in the observed signal, statistically independent of all $\{a_p\}$. Of particular interest in this formulation is the

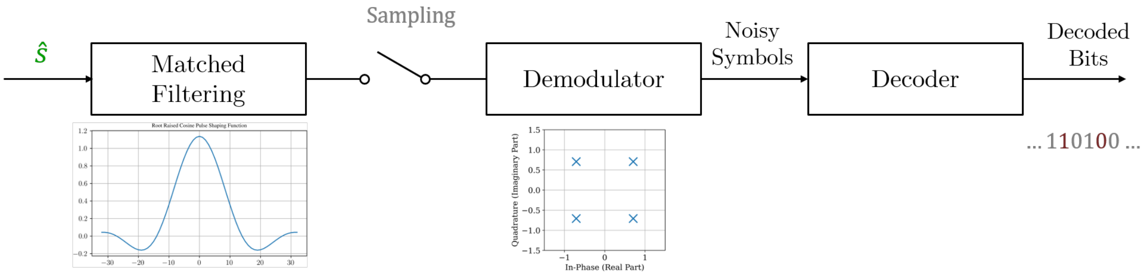


Figure 2-3: Block diagram of the matched filtering demodulation pipeline.

transmit pulse shaping function $g_{\text{tx}}(t)$, where we chose to use the RRC function.

At the receiver, we seek a receiver filter, $g_{\text{rx}}(t)$, such that the filtered and sampled output

$$y_{\text{filt}}(t) = \underbrace{g_{\text{rx}}(t) * g_{\text{tx}}(t)}_{:=g(t)} * \sum_p a_p \delta(t - pT_s) + g_{\text{rx}}(t) * w(t) \tag{2.6}$$

$$y[n] = y_{\text{filt}}(nT_s) = \sum_p q_p\, g((n-p)T_s) + \underbrace{\int w(\tau)\, g_{\text{rx}}(nT_s - \tau) d\tau}_{:=v[n]} \tag{2.7}$$

$$= \underbrace{c_n\, g(0)}_{:=y_s[n]} + \underbrace{\sum_{p \neq n} c_n\, g((n-p)T_s) + v[n]}_{:=y_v[n]} \tag{2.8}$$

would maximize the output SINR. In other words, we are looking to maximize

$$\text{SINR} = \frac{\mathbb{E}\left[|y_s[n]|^2\right]}{\mathbb{E}\left[|y_v[n]|^2\right]} = \frac{\mathbb{E}\left[|c_n|^2\right] |g(0)|^2}{\mathbb{E}\left[|c_n|^2\right] \sum_{p \neq n} |g(pT_s)|^2 + \sigma^2_{\text{AWGN}} \int |G_{\text{rx}}(f)|^2 df} \tag{2.9}$$

(where $G_{\text{rx}}(f)$ is the Fourier transform of $g_{\text{rx}}(t)$) via an appropriate choice of $g(t)$—and thereby, $g_{\text{rx}}(t)$. This can be done by finding an upper bound on the SINR that reaches equality for the appropriate filter choices. Ultimately, one such choice is $g_{\text{rx}}(t) = g^*_{\text{tx}}(-t)$—termed as the *matched filter*—that leads to a maximized SINR. In the case of an RRC pulse shaping function (which is real and symmetric), the matched filter is also the same RRC function.

In this work, we may refer to matched filtering more broadly to also include the detector/decoding step, as shown in Figure 2-3. As part of the demodulation pipeline, the filtered output is sampled (as in (2.8)), and then mapped to the closest symbol in a predefined constellation (in the Euclidean distance sense). Finally, we can map these complex-valued symbols back to their corresponding bits to recover the underlying information. We use this as a standard demodulation/detection pipeline for our RRC signal (where applicable).

Demodulation with matched filtering is optimal for waveforms in the presence of additive Gaussian noise. However, the core problem in this thesis is the presence of an additive interference that is not necessarily Gaussian. Hence, this work involves exploring pre-processing pipeline in the form of signal estimation and/or interference mitigation prior to the matched filtering/demodulation step.

## 2.4 Signal Estimation

The goal of signal estimation is to estimate the signal $s$ based on an observed signal $y$ corrupted by additive noise and/or interference, which we denote as $b$. Mathematically, the observation is expressed as

$$y = s + b,$$

where $y$ is the observed signal, comprising $s$ as the true signal-of-interest (SOI), and $b$ as the additive interference and/or noise.

Matched filtering, which was discussed above, can be viewed as a least-squares solution for estimating the SOI. If we assume the SOI to be a linearly modulated signal, i.e., that it can be represented as

$$s = H\,a,$$

where $H$ is the linear modulation operator (e.g., a matrix corresponding to the RRC pulse shaping function), and $a$ is the vector of complex-valued IQ symbols, given a known $H$ and assuming Gaussian statistics for $b$, matched filtering can be interpreted as a least squares estimation of the latent vector $a$, which can then be used to recover an estimate of $s$. However, in general, the interference $b$ is not necessarily Gaussian, resulting in a mismatch in its statistical model and thereby suboptimal performance in recovering $s$.

In the next subsection, we explore an alternative approach that accounts for the statistical properties of the interference in signal estimation.

### Minimum Mean Square Error Estimation

Minimum Mean Square Error (MMSE) estimation is a widely used technique in signal processing for estimating an unknown signal corrupted by interference and/or noise. It provides an optimal solution by minimizing the mean square error (MSE) between the estimated signal and the true signal.

MSE is a common metric used to quantify the quality of the estimated signal by measuring the average squared difference between the estimated signal and the true signal. The MSE is defined as

$$\text{MSE} = \mathbb{E}\left[\|x - \widehat{x}\|^2\right],$$

where $\widehat{x}$ is the estimated signal and $\mathbb{E}[\cdot]$ denotes the expectation operation.

The MMSE estimation aims to find an estimate $\widehat{x}$ that minimizes the MSE. Mathematically, (under finite mean and variance assumption on the signal components) the MMSE estimate $\widehat{x}_{\text{MMSE}}$ is given by

$$\widehat{x}_{\text{MMSE}} = \arg\min_{\widehat{x}} \mathbb{E}\left[\|x - \widehat{x}(y)\|^2\right] = \mathbb{E}[x|y]$$

where $\mathbb{E}[x|y]$ represents the conditional expectation of the unknown signal $x$ given the observed signal $y$.

Thus, the MMSE estimator minimizes the MSE, leading to accurate and reliable estimates in the mean-square sense. However, obtaining such an MMSE estimator can be complex. In some cases, such an estimator can be highly non-linear, and even analytically intractable.

For practical considerations, practitioners may look toward the linear MMSE estimator. In this case, we are interested in the MMSE estimator within a constrained family of linear operators, i.e., that

$$\arg\min_{W,d} \mathbb{E}\left[\|x - \widehat{x}(y)\|^2\right] \qquad \text{s.t.} \qquad \widehat{x}(y) = Wy + d, \tag{2.10}$$

where the parameters of the optimal linear estimator are given by

$$W = C_{XY}C_Y^{-1}\ ,\ d = \bar{x} - W\bar{y},$$

where $\bar{x} = \mathbb{E}[x]$, $\bar{y} = \mathbb{E}[y]$, $C_Y = \mathbb{E}[(y - \bar{y})(y - \bar{y})]$ is the autocovariance of $y$ and $C_{XY} = \mathbb{E}[(x - \bar{x})(y - \bar{y})]$ is the cross-covariance between $x$ and $y$.

A particular case in the discussion of MMSE estimation is when the signals under study are wide-sense stationary random processes, and we are interested in a linear, time-invariant filter for MMSE estimation. The optimal solution in such a case corresponds to the (non-causal discrete time) Wiener filter [51].

We remark that the linear MMSE estimator provides a practical solution in many scenarios, and is favored for its robustness and interpretability. Furthermore, in the absence of access to the true statistics of $x$ and $y$ (i.e., their first and second-order moments), we may adopt their corresponding empirical statistics by estimating from available data. Additionally, there is also a range of adaptive filtering algorithms, such as least-mean squares and recursive least-squares, aimed at arriving at the optimal linear estimator parameters in a data-driven fashion [52]. Nevertheless, it should be noted that better performance could be achievable through non-linear estimators.

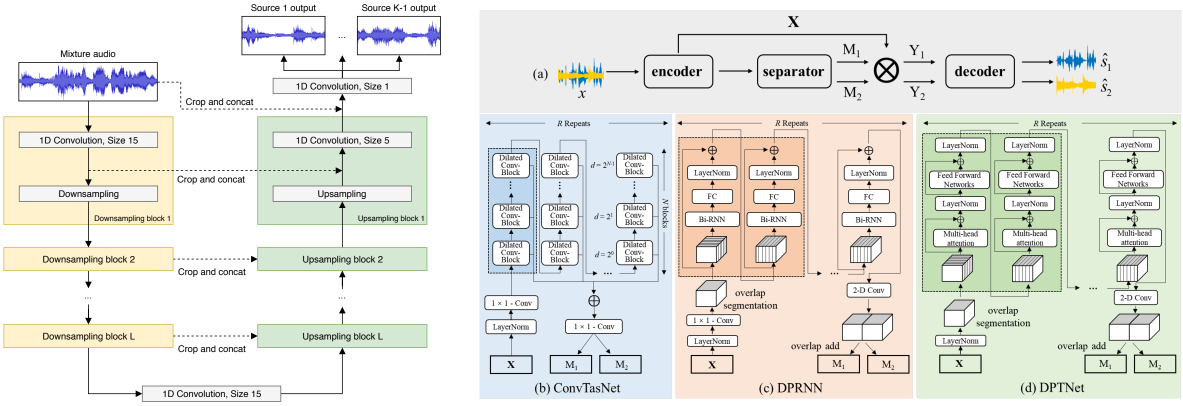
A key challenge so far is that the optimal (non-linear) MMSE estimator relies on known models of the signal components—which may not be available in practice. Instead, it might be more common to be provided with examples, from which we seek to learn the underlying characteristics and thus the associated MMSE estimator. This task could be achievable through data-driven approaches and machine learning, which is the main focus of this thesis.

# Chapter 3

# On Neural Architectures for Separating OFDM Waveforms

In this section, we delve into a problem abstraction centered around OFDM waveforms, which are pertinent to modern RF systems. Broadly, we consider a seemingly simple scenario, where if the signal models were known, we might expect a relatively simple solution. Yet surprisingly, the generic application of deep learning techniques fails to perform well. To set the stage, it is worth noting the significant strides made in the recent decade with deep learning methods for source separation, predominantly focusing on separating image or audio sources, as discussed in Section 1.2. Notably, for single-channel *time-domain* audio separation, state-of-the-art solutions have benefited from novel neural architectures.

One of the earlier works in deep learning for audio source separation, particularly operating on the raw waveform (1-dimensional time domain representation), is the Wave-U-Net [23]. This is based on a U-Net structure [54]—a fully convolutional neural network comprising successive downsampling and upsampling blocks with skip connections—with key modifications introduced to learn an interpolation function between upsampling blocks. More contemporary methods adopt a TasNet structure [30], encompassing an encoder, a decoder, and a separator block. Functionally, the encoder block transforms the input to a latent feature space, for which the separator forms a mask on the latent representation; the decoder block subsequently transforms the separated latent representation to the raw waveform. Recent efforts revolve around new architectures for these blocks for more effective and efficient separation of audio signals. For example, modifications that introduce convolutional, recur-



(a) Wave-U-Net

(b) Separation neural network architectures with an encoder-separation-decoder framework

Figure 3-1: Selected neural architectures used in audio separation. (a) Figure is from [23], showing an illustration of the Wave-U-Net architecture. (b) Figure is from [53], showing an illustration of the neural architecture used in audio separation. Common to these three architectures are the encoder-separation-decoder framework, building on TasNet architecture [30].

rent, and attention-based layer structures to the separation network have been introduced, with varying degrees of success for the audio separation problem [31–34]. Visualizations of these neural architectures are presented in Fig. 3-1.

Implicit in these methods are strategies to exploit the properties of typical audio signals. In fact, it is believed that the features exploited by state-of-the-art neural architectures are related to separability in the time-frequency space [53].

Similar to audio separation in the time domain, we are interested in separating RF signals which are 1-dimensional time series. Therefore, one might be inclined to use state-of-the-art methods from audio separation to separate RF signals. Yet, the properties of RF waveforms differ from audio signals—e.g., RF communication waveforms tend to pack a large amount of information into a finite frequency band, rendering them no longer sparse in the time-frequency space. In fact, signals may overlap in this space, a condition known as "co-channel". The critical challenge is the separation of co-channel signals, in which the sources overlap, partially or fully, in *both* time and frequency.

Given that the characteristics of the underlying sources under consideration differ for the respective domains, this would lead to different behaviors and performances. And while audio-oriented neural networks can work on time series inputs and have been shown to be successful with other modalities (e.g., with seismic signals [35]), it is uncertain if the same neural architectures are also effective at separating RF communication signals. A different signal modality typically requires accompanying innovations or the discovery of appropriate machine learning model architectures that can best capture their characteristics. We are interested in what these correspond to in the space of RF waveforms.

This chapter empirically demonstrates the limitations of existing methods and proposes a signal-processing methodology for neural architectural choices in the context of single-channel source separation for OFDM signals. In particular, we consider a prototype problem based on the OFDM model, posed such that perfect separation of the signals is technically attainable with prior knowledge of the signal model, but challenging without it. Under this setup, we study whether neural network-based approaches can learn to exploit the underlying OFDM structures for signal separation.

To the best of our knowledge, this work is the first to assess the performance (and therefore, the ineffectiveness in some regimes) of neural architectures from audio separation when applied to OFDM waveforms, serving as an important benchmark. We also propose modifications, inspired by OFDM structures, that significantly improve separation performance. The key takeaways are the distinct challenges posed by digital communication signals for existing neural methods in signal separtion, and judicious adaptations to advance neural methods for time-domain signals beyond the efforts in the audio domain.

## 3.1 Motivation: Learning to Separate OFDM Waveforms

The key area of interest in this chapter is digital signals that use OFDM, a digital modulation scheme widely used in modern wireless protocols, such as WiFi and 4G/LTE/5G [55]. A defining feature of OFDM waveforms is the encoding of information on orthogonal sinusoids (also termed subcarriers), allowing for more efficient packing of information bits within a given bandwidth. Decoding OFDM signals typically involves the extraction of the encoded bits from these subcarriers, which can be done using efficient algorithms like the FFT. Nonetheless, it is vital to know the model parameters, such as the subcarrier spacing (also referred to as FFT size); a mismatch in the parameter estimation results in a loss of orthogonality among the estimated subcarriers, resulting in a suboptimal recovery of the underlying data.

We also empirically observe OFDM waveforms as a challenging class of signals to tackle in data-driven single-channel source separation [40, 41]. Particularly, as highlighted in [42] and referring to the mathematical description of an OFDM waveform in Section 2.2.1, each time-domain sample is a sum of statistically independent random variables, and hence asymptotically Gaussian as the number of subcarriers $L \to \infty$; similar characterization holds true within a fixed window of $W \ll L$ time samples (unless the samples are exactly a period away) [9]. This implies that, when only considering the "local" features in such a time series, it may appear as if treating an OFDM signal as an additive white Gaussian noise is the best approach. Said differently, a carefully chosen algorithmic architecture is critical to uncover the underlying non-Gaussianity that can be exploited for better signal separation performance in this problem. This part of the thesis primarily concerns OFDM signals and the relevant deep-learning architectural choices that would help capture their characteristics.

In this chapter, we focus on source components represented by OFDM waveforms to examine whether machine learning methods can effectively capture and utilize features specific to OFDM signals for signal separation. To address this, we design an abstraction of the problem setup in a way such that the signals are inherently separable at the subcarrier level; however, the extraction of these subcarrier symbols becomes challenging without explicit knowledge of the OFDM source model parameters. Through this analysis, we aim to gain insights into the architectural choices of machine learning methods that can effectively learn and leverage the underlying OFDM structures, even in the absence of explicit signal model parameters, thereby improving signal separation performance.

## 3.2 Related Works on Deep Learning with OFDM Waveforms

Related works on deep learning for time series have been discussed earlier in Section 1.2. This section reviews related works in signal processing and data-driven methods involving OFDM waveforms.

Much effort has been in the synchronization and parameter estimation of OFDM waveforms. There are several directions on this front. One of the key approaches is by using some deterministic, known properties of the OFDM waveform, such as pilots embedded in selected subcarriers [56, 57]. Another approach reported in a related work involves a handcrafted method in determining the subcarrier spacing and cyclic prefix length based on common specifications of OFDM waveforms [58, 59]. However, generally, these depend on certain explicit properties of current OFDM waveforms, and may be inapplicable when the specifications or the receiver characteristics depart from the assumptions outlined.

Most recently, neural network methods relating to end-to-end training of OFDM receivers have been discussed. These neural network methods are effective in capturing more complex phenomenon such as multi-path fading and other channel effects [47, 60, 61], and facilitates improved demodulation and decoding performance. Nevertheless, it seems that many of these methods still require knowledge about the OFDM parameters and/or the ability to synchronize to the OFDM symbol start time (which is assumed to be available under these settings) to process the orthogonal subcarriers. Particularly, many of these methods contain an FFT operation, requiring the actual FFT size of the OFDM waveform. If we do not have knowledge about these OFDM parameters, as in our problem of source separation in the absence of source models, these methods are of limited applicability.

In the realm of deep learning approaches to single-channel source separation of RF signals, it is worth noting that related works have been predominantly concentrated on single-carrier signals and radar signals [36–39]. Notably, apart from our recent works [40–42], we are not aware of any other model-blind signal separation approaches involving mixtures with OFDM signals.

## 3.3 Problem Formulation

We reframe the signal separation problem with slight modifications to our formulation and terms. Consider an observed $N$-length 1-dimensional signal

$$\boldsymbol{y} = \boldsymbol{s} + \boldsymbol{b}, \tag{3.1}$$

where $\boldsymbol{s} \triangleq [s[0] \, \dots \, s[N-1]]^{\mathrm{T}} \in \mathbb{C}^N$ is our signal-of-interest (SOI) to be extracted, and $\boldsymbol{b} \triangleq [b[0] \, \dots \, b[N-1]]^{\mathrm{T}} \in \mathbb{C}^N$ is the interference (signal-not-of-interest). The goal is to separate $\boldsymbol{s}$ from $\boldsymbol{b}$—or equivalently, to estimate $\boldsymbol{s}$ from $\boldsymbol{y}$—with minimum mean squared error (MSE) as the criterion. We assume that the models for $\boldsymbol{s}$ and $\boldsymbol{b}$ are not known; however, we have access to a dataset of $M$ i.i.d. examples, $\{(\boldsymbol{y}^{(i)}, \boldsymbol{s}^{(i)})\}_{i=1}^{M}$.

One caveat in our problem abstraction is that we consider a greatly simplified scenario involving fewer parameters that describe the source signals. (We will discuss the constraints of these modeling choices, as well as avenues for broadening the scope and generalization in future works, towards the end of this chapter.) However, based on our observations and empirical evidence, even within the context of this problem abstraction, we encounter cases that present non-trivial challenges. In this work, we consider an SOI and interference that are discrete-time OFDM waveforms, formally expressed as

$$\begin{aligned} s[n] &= \sum_{p=0}^{P-1} \sum_{k=0}^{K-1} g_{k,p}\, r[n - p \cdot (K + T_{\text{cp}}) - T_{\text{cp}},\, k], \\ b[n] &= \sum_{p=0}^{P-1} \sum_{k=0}^{K-1} h_{k,p}\, r[n - p \cdot (K + T_{\text{cp}}) - T_{\text{cp}},\, k], \\ r[n,\, k] &\triangleq \exp(j2\pi kn/K)\, \mathbb{1}_{\{-T_{\text{cp}} \leq n < K\}}, \end{aligned} \tag{3.2}$$

for $n \in \{0, \ldots, N-1\}$, where $K \in \mathbb{N}$ is the total number of orthogonal complex sinusoid terms (also termed as subcarriers). Note that $K$ also corresponds to the FFT size, and is typically chosen to be an even number. The coefficients $g_{k,p} \in \mathcal{G}$, $h_{k,p} \in \mathcal{H}$ are the modulated symbols, and $\mathcal{G}, \mathcal{H}$ are their alphabets (constellations), respectively. A cyclic prefix (CP) is typically added before an OFDM symbol, which is a short replica prepended to serve as a guard interval from the previous symbols. Hence, each OFDM symbol is described for the interval $[-T_{\text{cp}}, K]$ (as expressed by the indicator function $\mathbb{1}_{\{\cdot\}}$), where $T_{\text{cp}} \in \mathbb{N}$ is the CP length, and $K$ is the OFDM symbol length (without CP). The signals span $P$ OFDM symbols, and their individual finite support is reflected by the finitely supported function $r[n,\, k]$. We remark that this scenario can be viewed as a multiple-access setup involving two coordinated OFDM sources. This particular setup simplifies the problem, making it more amenable for investigation and analysis.

In this setting, the observed mixture can also be viewed as an OFDM waveform, with the coefficients being elements from the superconstellation of the SOI's and interference's symbols, i.e., the Minkowski sum $\mathcal{A} \triangleq \mathcal{G} \oplus \mathcal{H}$, such that

$$\begin{aligned} y[n] &= \sum_{p=0}^{P-1} \sum_{k=0}^{K-1} a_{k,p}\, r[n - p \cdot (K + T_{\text{cp}}) - T_{\text{cp}},\, k], \\ a_{k,p} &= g_{k,p} + h_{k,p}, \quad a_{k,p} \in \mathcal{A}. \end{aligned} \tag{3.3}$$

The existence of a surjective function $f : \mathcal{A} \to \mathcal{G}$, i.e., every element in $\mathcal{A}$ can be uniquely associated with an element in the SOI's constellation $\mathcal{G}$, suffices for perfect separability.[1]

[1] Alternatively, a surjective function $f : \mathcal{A} \to \mathcal{H}$.

### Special Case: Real-valued OFDM Signals

To prioritize the fundamental elements of this problem, namely the underlying Fourier structures and finite coefficient sets present in OFDM waveforms, we propose to simplify the source models further and reduce the complexity/number of parameters in the following special case. Consider (3.2) with $P = 1$, $T_{\mathrm{cp}} = N - K$ and $N \in K \cdot \mathbb{N}$, namely,

$$s[n] = \sum_{k=0}^{K-1} \underbrace{g_{k,0}}_{\triangleq g_k} r[n - T_{\mathrm{cp}}, k] = \sum_{k=0}^{K-1} g_k \exp(j2\pi kn/K), \tag{3.4}$$

and similarly, for $b[n]$, $y[n]$ with coefficients $h_k$, $a_k$, respectively, such that the (periodic extensions of the) SOI and interference are discrete Fourier series. Further, we impose the conjugate symmetry constraint on the coefficients $g_k$, $h_k$,

$$\text{①}\ g_0 = g_{K/2} = 0 \quad \text{②}\ g_k = g^*_{K-k}, \forall k \in \{1, \ldots, \tfrac{K}{2} - 1\},$$

and similarly for $h_k$, where $z^*$ denotes the complex conjugate of a complex number $z$. Consequently, the waveforms generated by (3.4) are real-valued, i.e., $\boldsymbol{s}, \boldsymbol{b} \in \mathbb{R}^N \Rightarrow \boldsymbol{y} \in \mathbb{R}^N$.

The rationale behind examining this particular case is to evaluate the ability of candidate neural architectures to effectively capture or learn to leverage the inherent *orthogonality* of subcarriers and the *discrete* constellation set. We posit that perfect separation becomes theoretically achievable should these characteristics be appropriately captured by the separation method/neural architecture.

## 3.4 Model-Based Insights to the Problem

To gain insights into the separation of OFDM signals, we begin by examining the setup in (3.4) and referring to conventional model-based approaches as a point of reference. Typically, one would consider looking at the frequency spectrum by performing a Fourier transform, aiming to extract the SOI from the estimated coefficients of the frequency spectrum. This process necessitates a sufficiently large FFT size to preserve the orthogonality of the subcarriers. If the FFT size is insufficient, spectral leakage from neighboring subcarriers occurs, leading to a significantly larger superconstellation and loss of orthogonality (as depicted in

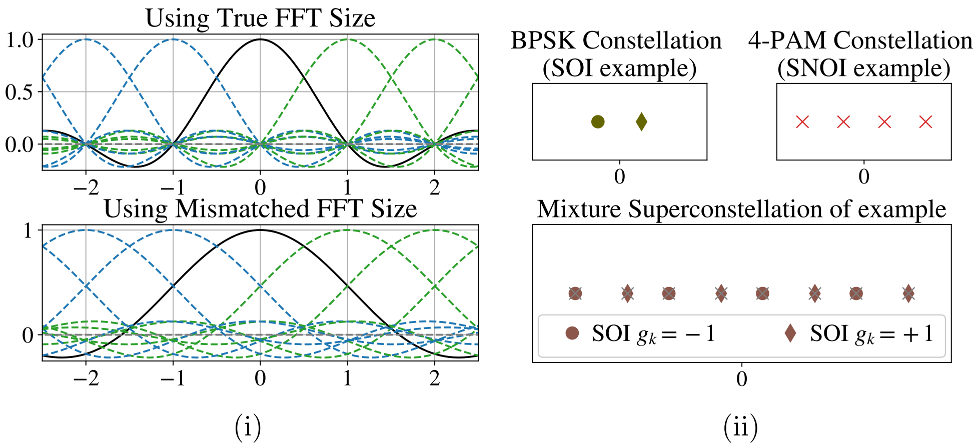


Figure 3-2: Visualization of OFDM structure—(i) using the appropriate FFT size leads to orthogonality between subcarriers; a mismatched FFT leads to a loss of orthogonality at the subcarrier frequencies; (ii) for an appropriate choice of discrete constellations, a surjective mapping of points from the superconstellation to an SOI symbol can be obtained.

Fig. 3-2(I)).[2] Subsequently, establishing a data-driven or handcrafted transformation of the frequency spectrum, corresponding to the subjective mapping of the mixture's symbol superconstellation to the SOI's constellation points (e.g., Fig. 3-2(ii)), is required. While such a routine demonstrates a potential approach to achieve perfect signal separation, it may not be practicable in more general scenarios (e.g., in (3.2), where the time offset and FFT size choice are critical considerations).

Furthermore, we observe that through this framework, the MMSE would be 0. Recall that the MMSE estimator can be expressed as

$$\widehat{\boldsymbol{s}}_{\mathrm{MMSE}}(\boldsymbol{y}) = \mathbb{E}\left[\boldsymbol{s}|\boldsymbol{y} = \boldsymbol{s}^* + \boldsymbol{b}^*\right]$$

where $\boldsymbol{s}^*$,$\boldsymbol{b}^*$ denote the ground truth latent sources; and that, having observed $\boldsymbol{y}$, the probability mass function[3] $P(\boldsymbol{s}|\boldsymbol{y})$ is a Kronecker delta concentrated around the true $\boldsymbol{s}^*$ found in $\boldsymbol{y}$. Consequently, we get that $\widehat{\boldsymbol{s}}_{\mathrm{MMSE}}(\boldsymbol{y}) = \boldsymbol{s}^*$ (since it is uniquely determined upon observing $\boldsymbol{y}$), resulting in an effective error of 0. Again, while such an MMSE estimator exists in theory, this does not translate to an implementable algorithm without explicit knowledge about the signal model parameters.

[2] We only require a sufficiently large FFT size in this highly oversampled case. It is important to note that, in practice,for cases like (3.2), an *exact* FFT size is actually required to avoid intersymbol interference and ensure accurate recovery of the subcarrier symbols.

[3] This statement is specific to (3.4) where the coefficients are from a discrete constellation; as a result, there is only a finite number of realizations for $\boldsymbol{s}$ and $\boldsymbol{b}$, and thereby also for $\boldsymbol{y}$.

In the next section, we explore the potential of neural networks as function approximations, aiming to learn a function that is akin to the pipeline described above, or at least one that achieves similar performance. Specifically, by training a neural network to minimize MSE, we hope to obtain a function approximator of the MMSE estimator, for which the error would be close to 0—up to limitations in numerical methods and approximations.

## 3.5 End-to-end Separator via Deep Neural Networks

State-of-the-art solutions for source separation of audio signals in the time domain benefit from deep learning approaches, many of which propose novel neural architectures to achieve improved separation ability. Fig. 3-1 shows some of these neural architectures from recent time-domain audio separation works, which also correspond to the architectures we use later in our computational simulations. Implicit in these methods are strategies to exploit the properties of typical audio signals. In fact, it is believed that the features exploited by state-of-the-art neural architectures—particularly by the encoder-separator-decoder framework—are related to separability in the time-frequency space [53].

On the other hand, these neural network methods from audio separation do not require explicit information about the source models. The only practical constraint is that existing implementations are made for real-valued time series inputs. By considering the special case established in (3.4) with real-valued latent sources, we can naturally adopt the neural architectures proposed in audio source separation works, and assess their effectiveness to our problem.

The neural network methods used in this context do not explicitly leverage information about the sources being an OFDM waveform or discrete Fourier series. However, an effective architecture for this signal separation problem ought to be capable of learning and exploiting the inherent properties of OFDM, such as its subcarrier structure and discrete symbol constellations. Surprisingly, we find that, beyond a limited regime of this problem, conventional audio-based neural architectures fail to separate OFDM mixtures that are inherently perfectly separable. To address this shortcoming, we later propose domain-informed modifications to these architectures, leading to successful separation and significant performance improvements, in terms of MSE, by orders of magnitudes.

## 3.6 Computational Simulations

For the empirical aspect of this work, we primarily look at computational experiments with parameters $N = 4096$, $K = 64$, $K_{\mathrm{sc}} = 28$, where $K_{\mathrm{sc}}$ corresponds to the number of unique nonzero coefficients (subcarriers) in this model; these parameters are, in part, based on 802.11n WiFi waveform properties [62]. We consider 4 different cases of $g_k$, $h_k$, for $k \in \{1, \ldots, K_{\mathrm{sc}}\}$:

- **Case 1: Disjoint frequency sets:** $g_k = 0$ when $h_k \neq 0$ and *vice versa*, where nonzero indices are randomly chosen once, and stay fixed thereafter. The nonzero coefficients are drawn from a random continuous uniform distribution, $g_k \sim \mathcal{U}[-\sqrt{3}, \sqrt{3}]$, $h_k \sim \mathcal{U}[-4\sqrt{3}, 4\sqrt{3}]$.

- **Case 2: "BPSK[4]-like" coefficients:** $g_k \in \{+1, -1\}$ and $h_k \in \{+4, -4\}$.

- **Case 3: "Mixed" coefficients:** $g_k \in \{+1, -1\}$ and $h_k \in \{{}^{+12}/\sqrt{5}, {}^{+4}/\sqrt{5}, {}^{-4}/\sqrt{5}, {}^{-12}/\sqrt{5}\}$.

- **Case 4: "4-PAM[4]-like" coefficients:** $g_k \in \{{}^{+3}/\sqrt{5}, {}^{+1}/\sqrt{5}, {}^{-1}/\sqrt{5}, {}^{-3}/\sqrt{5}\}$ and $h_k \in \{{}^{+12}/\sqrt{5}, {}^{+4}/\sqrt{5}, {}^{-4}/\sqrt{5}, {}^{-12}/\sqrt{5}\}$.

The appropriate scaling factors on the source components are introduced such that the SOI $\boldsymbol{s}$ has unit average power, and that the average interference power is 16 times that of the average SOI power (i.e., corresponding to a SIR of $-12.041$ dB). We set $g_0 = h_0 = 0$, and $g_k = h_k = 0$ for $k \in [K_{\mathrm{sc}} + 1, K/2]$, and recall that for $k > K/2$, the coefficients are constrained to have conjugate symmetry.

Recall that the above sets of coefficients were chosen so that separability of the waveforms could be achieved by exploiting the orthogonality of the complex sinusoids and learning the surjective mapping of the coefficients. If we have prior knowledge of the source model, i.e., the frequencies of cosines present, the problem can be approached by performing a maximum likelihood estimation on $\{g_k\}$, $\{h_k\}$, allowing us to reconstruct the SOI waveform. The challenge, once again, lies in the absence of source models—notably, we assume no knowledge about the frequency spacing and the set of coefficients described above. Instead, one has to learn the above model from the available data.

[4] BPSK: Binary Phase Shift Keying; 4-PAM: 4 Pulse-Amplitude Modulation; these are modulation schemes typical in digital communication signals.

Table 3.1: MSE (in decibels, dB) of the extracted SOI using audio-domain neural networks. Entries with MSE$< 10^{-2}$ (i.e., $-20$ dB[6]) are in red.

| | Case 1<br>Disjoint | Case 2<br>BPSK+BPSK | Case 3<br>BPSK+4-PAM<br>(Mixed) | Case 4<br>4-PAM+4-<br>PAM |
|---|---|---|---|---|
| Wave-U-Net [23] | $-57.246$ dB | $-46.827$ dB | $-4.663$ dB | $-4.665$ dB |
| Conv-TasNet [31] | $-40.790$ dB | $-12.179$ dB | $-1.060$ dB | $-1.009$ dB |
| Sudo-Rm-Rf [32] | $-37.023$ dB | $-26.493$ dB | $-12.855$ dB | $-11.495$ dB |
| Dual Path RNN [33] | $-41.425$ dB | $-27.302$ dB | $-0.671$ dB | $-0.542$ dB |
| DPTNet [34] | $-36.825$ dB | $-33.652$ dB | $-3.548$ dB | $-2.432$ dB |

To the best of our knowledge, we have not found established baseline methods for the problem formulation established ((3.3) with (3.4)). Hence, part of our work is to train selected state-of-the-art neural networks from audio separation [23, 31–34]—demonstrated to be effective in audio source separation in their respective works—for this problem, and assess their performance to serve as our comparison benchmark.

Asteroid, the PyTorch-based audio source separation toolbox [63], is used for state-of-the-art audio separation neural architectures, whereas Wave-U-Net and its modified version (our proposed architecture, detailed later) are implemented in PyTorch.[5] We use $90,000$ and $10,000$ independent realizations of mixture-SOI pairs for the training and validation sets respectively. Adam optimizer with a learning rate $10^{-4}$ is used to train the respective neural networks for $2,000$ epochs, with early stopping after 100 epochs of no improvement on the validation set. Table 3.1 reports the MSE performance of the selected neural architectures in the reconstruction of the SOI, on a separately generated, unseen test set comprising $1,000$ examples. We train each neural network on a computing node from a high-performance computing cluster with Intel Xeon Gold 6248, 192 GB RAM, and an NVidia Volta V100 GPU.

Unsurprisingly, the audio-domain neural network models are all good in separating Case 1 with disjoint frequencies, i.e., identifying and filtering the frequencies that make up the SOI $\boldsymbol{s}$. One should note that while perfect separation is theoretically possible, the neural network, as function approximation, might induce some numerical issues and approximation errors; nevertheless, having a resulting MSE of less than $-20$ dB after separation (from an original

[5] Repository containing code and implementation details: https://github.com/RFChallenge/SCSS_OFDMArchitecture.

[6] An approximation of the best separation performance reported in [23, 31–34].

unmitigated MSE of around 12 dB) can be sufficiently beneficial in many applications.

On the other hand, the other cases appear to be deceptively simple tasks, and justifiably so since perfect separation can be achieved by relatively simple operations—i.e., a linear transformation like the FFT, followed by a mapping function for the coefficients. Yet, the characteristics of such a mixture differ significantly from those typically encountered in audio mixtures. Specifically, the source components are co-channel, and they are not sparse in the time-frequency space; thus, a simple filter or spectrogram-based masking would not suffice in separating these signals. Further, note that we are interested in signals with very similar correlational structures in time, making it a distinct problem from what is outlined in the previous section. In this case, the non-Gaussianity of the sources has to be exploited for an effective source separation; notably, the FFT size and the discrete nature of the coefficients underlying their generative processes are essential in a model-based approach to this problem; we are interested in whether learning-based approaches are able to learn representations related to these, and achieve good separation performance simply by learning from examples. Thereafter, we review possible justifications for the improvement attained by drawing connections to OFDM's Fourier structures, which in turn leads to guidelines for domain-informed parameterization.

A key goal of this is to explore neural network architectures that can capture the underlying subcarrier structures. This preliminary experiment exposes the shortcoming of state-of-the-art neural network efforts when applied to RF waveforms, and particularly by a simplified abstraction of the OFDM waveform.

In Case 2, where co-channel sources are considered, most of the audio-domain neural network architectures demonstrate good performance in separating these signals. On the other hand, the signal separation performance significantly deteriorates in Cases 3 and 4. The models' success in Case 2 suggests that the separation mechanisms employed by these audio separation methods have the capability to separate co-channel signals, i.e. deviating from the proximity characteristics indicated in the comparison study [53]. This reflects the potential generalizability of some of these architectures. Nonetheless, Cases 3 and 4 present a more challenging scenario that may require a more complex mapping function for effective separation. This could be a breaking point for these audio-based architectures, highlighting the need for alternative approaches in these cases.

Table 3.2: MSE (in dB) of the extracted SOI using our proposed architecture, the modified Wave-U-Net. Improvement of the modified Wave-U-Net, compared to the best-performing benchmark method, is reported in parenthesis in the first row. The MSE achieved by other audio-based separator neural networks is included again for ease of comparison. Entries with MSE$< 10^{-2}$ (i.e., $-20$ dB) are in red.

| | Case 1<br>Disjoint | Case 2<br>BPSK+BPSK | Case 3<br>BPSK+4-PAM<br>(Mixed) | Case 4<br>4-PAM+4-PAM |
|---|---|---|---|---|
| **Modified Wave-U-Net (Proposed)** | **$-65.526$ dB**<br>($\downarrow$ 8.280) | **$-47.558$ dB**<br>($\downarrow$ 0.731) | **$-47.377$ dB**<br>($\downarrow$ 34.522) | **$-41.156$ dB**<br>($\downarrow$ 29.661) |
| Wave-U-Net [23] | $-57.246$ dB | $-46.827$ dB | $-4.663$ dB | $-4.665$ dB |
| Conv-TasNet [31] | $-40.790$ dB | $-12.179$ dB | $-1.060$ dB | $-1.009$ dB |
| Sudo-Rm-Rf [32] | $-37.023$ dB | $-26.493$ dB | $-12.855$ dB | $-11.495$ dB |
| Dual Path RNN [33] | $-41.425$ dB | $-27.302$ dB | $-0.671$ dB | $-0.542$ dB |
| DPTNet [34] | $-36.825$ dB | $-33.652$ dB | $-3.548$ dB | $-2.432$ dB |

## 3.7 Insights on OFDM Domain-informed Architecture

We now propose modifications to one of the architectures based on our earlier insights from OFDM signals. Thereafter, we review possible justifications for the improvement attained by drawing connections to OFDM's Fourier structures, which in turn leads to guidelines for domain-informed parameterization.

### 3.7.1 Proposed Neural Architecture Modifications

Referring to the model-based approach, we seek a neural network that is capable of approximating an appropriately sized FFT operator (Fig. 3-2). Based on this insight, a natural modification to the neural network is to increase the number of filters (20× as many) and the receptive fields of these filters on the first layer (kernel size $W = 101$), which operates on the time-domain input. We introduce these modifications to Wave-U-Net—the simplest among those investigated. The first row of Table 3.2 reports the substantial improvement in MSE due to these modifications.

To further lend credence to the role of first-layer kernel size, we show the signal separation results on Case 4 using the modified Wave-U-Net with different sizes in Table 3.3. Here, we see a significant improvement in separation performance when kernel sizes 63 and longer are used in the modified Wave-U-Net (recalling that the true FFT size $K = 64$).

Table 3.3: MSE (in dB) of the extracted SOI using the modified Wave-U-Net with different first-layer kernel sizes. Entries with MSE$< 10^{-2}$ (i.e., $-20$ dB) are in red to highlight the transition at $W \approx K_{\mathrm{sc}}$.

| Kernel Size | MSE | Kernel Size | MSE |
|---|---:|---|---:|
| $W = 15$ | $-6.030$ dB | $W = 65$ | $-42.824$ dB |
| $W = 21$ | $-5.621$ dB | $W = 71$ | $-42.099$ dB |
| $W = 31$ | $-6.183$ dB | $W = 81$ | $-42.690$ dB |
| $W = 51$ | $-16.319$ dB | $W = 101$ | $-41.156$ dB |
| $W = 63$ | $-41.380$ dB | $W = 201$ | $-44.319$ dB |

### 3.7.2 Features of Long 1st-Layer Convolutional Kernels

The experiments conducted so far have demonstrated the effectiveness of 1D convolutional filters with long kernel sizes in capturing relevant structures. However, it is important to delve deeper into why these filters work and understand the significance of our proposed modification. This section aims to provide an interpretation of the observed results and establish a broader guiding principle for choosing neural architectures and parameterization in this domain.

To begin, we revisit the model-based insights discussed in Section 3.4 and the importance of the FFT size. Referring to (3.4), each time-domain sample is a sum of statistically independent random variables; by the central limit theorem (CLT), each of these samples is marginally Gaussian distributed as $K \to \infty$. It is also worth noting that even for a fixed-length window of $W \ll K$, these time samples are asymptotically jointly Gaussian, unless they are exactly a period away [9]. Yet, $K$ consecutive time-domain samples are not jointly Gaussian, as evident from the discrete (non-Gaussian) coefficient set (where the CLT cannot be invoked in this case). A large receptive field in the neural network, particularly on the raw input, may be related to the window (FFT) size that allows capturing the underlying non-Gaussianity.

While it is difficult to provide a semantic interpretation as to why the modified Wave-U-net neural network performs well while the other architectures fail, we can gain insights by examining the learned Wave-U-Net through visualizations of its first-layer kernel weights. Fig. 3-3 presents a curated selection of the kernel weights, some of which resemble sinusoidal patterns of different frequencies. It is important to note that sinusoids with a resolution of $1/T$ can only be reliably represented by a segment no shorter than $T$. Hence, it appears that choosing longer windows enables the first-layer convolutional kernels to represent sinusoids

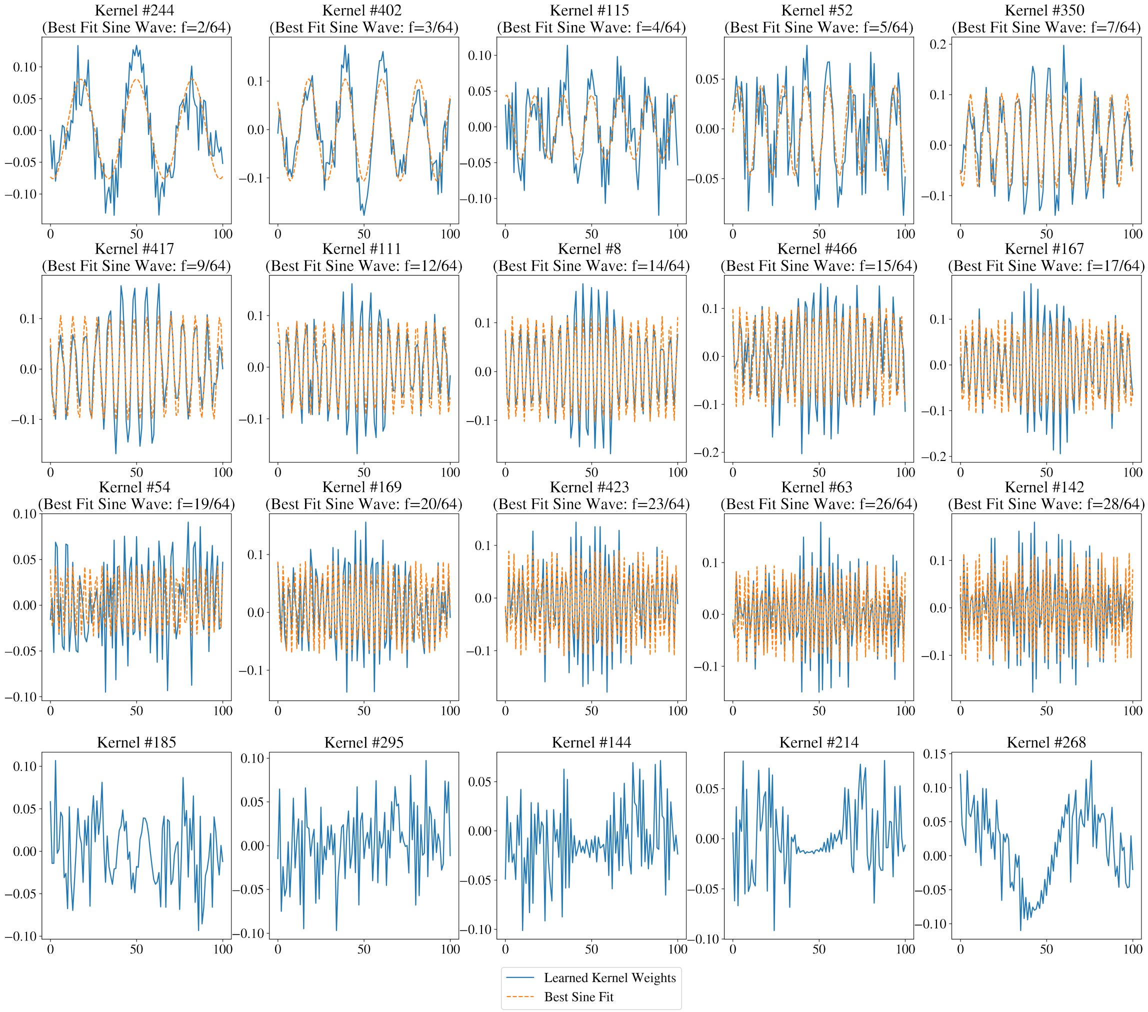


Figure 3-3: Features of a selected subset of the kernel weights on the first convolutional layer. For the first 15 kernel weights, we noticed patterns resembling sinusoids of different frequencies. The best fit sinusoidal fit is also included and overlaid in these visualizations.

of such a resolution. In other words, the first layer seems to be approximating some form of FFT of the appropriate size/frequency resolution, and this requires sufficiently numerous filters of a sufficiently long kernel length to effectively represent such information.

These findings reflect the value of selecting neural architectures judiciously, such as choosing a larger receptive field and an appropriate number of filters, as these factors play an essential role in achieving effective signal separation. We recognize that a deep neural network ought to have a large effective receptive field through its stacked layers, even if the kernel sizes of individual convolutional layers are short [64]. Yet, we have observed that none of the deep neural network models considered are as effective in Case 4, in contrast to what is achieved through a significantly long kernel on the first layer, operating *directly* on the input itself. The discrepancy in performance raises further questions and calls for deeper investigation.

## 3.8 Concluding Remarks

In this chapter, we have presented a simplified abstraction to assess the effectiveness of deep learning methods in signal separation involving OFDM structures. Our findings emphasize the critical role of thoughtful consideration and careful design when applying deep learning tools to this domain. Naively applying these methods to even such a simple problem yields suboptimal results, underscoring the existing gap in this field and the need for further exploration and refinement.

Moving forward, future work entails delving deeper into the general model formulation outlined in (3.2) and investigating the underlying mechanisms of the proposed modified Wave-U-Net architecture. Such an investigation will shed light on designing more effective neural network architectures capable of handling complex OFDM waveforms with larger symbol constellations, different FFT sizes, or variable cyclic prefix lengths.

In practical scenarios, we often lack complete knowledge of the true source model, including potential deviation from the model described in (3.2). This is particularly relevant in scenarios like the RFChallenge [22], where RF signals may exhibit OFDM characteristics but with unknown specifications and deviations due to hardware effects. Therefore, we aim to identify structural priors or guiding principles for neural architectures that are well-suited for these RF signals.

Reflecting on the problem abstraction adopted in this section, we recognize the particular assumptions made regarding the two signal components. Specifically, we assume that we have two OFDM waveforms that are perfectly synchronized and locked in time, frequency, and phase. Such scenarios could occur when wireless devices transmit OFDM signals in a coordinated multiple-access manner and are almost equidistant or co-located. Despite this narrow scope, our primary objective in this chapter remains to identify a simple configuration with a small number of parameters that is still representative of RF signal mixtures. This approach facilitates the evaluation of the efficacy and limitations of deep learning techniques, particularly those that have been developed for audio signals, when applied to the RF domain. Looking ahead, we are motivated to investigate beyond generalizations of the OFDM model, and to venture into addressing problems involving uncoordinated signal mixtures of different wireless devices.

Building on these insights, the subsequent chapters will explore the more realistic scenario of mixtures containing different signal types, as outlined in our original problem formulation (1.1). In this context, achieving the optimal separation performance may not be as straightforward (and no longer necessarily perfectly separable), and we aim to explore the achievable MMSE and develop strategies to improve separation performance accordingly.

# Chapter 4

# Benchmarking with Cyclostationary Gaussian Signals

This chapter focuses on the separation problem involving the class of *cyclostationary* signals—i.e., signals with periodically repeating statistical structures, which are particularly relevant to modeling RF communication and sensing systems (e.g., [65]). Interestingly, under some conditions, perfect separation of cyclostationary signals could be achieved despite having components with overlapping time-frequency spectra [66, 67]. More broadly, the theory of Wiener filter can be generalized for cyclostationary processes, corresponding to an optimal frequency shift filtering to separate temporally and spectrally overlapping signals [9, 68]; however, this requires precise knowledge or estimation of the underlying cyclostationary statistics. In practical scenarios, the challenge lies in accurately modeling the cyclostationarity of the signals without explicit knowledge of the underlying signal model.

The practical challenge lies in modeling such a cyclostationarity in a way that adequately captures the true statistics of the observed mixtures. A more realistic scenario is one where the signal model is unknown, but examples of the signals (by measurements or generative simulations[1]) are available. Furthermore, even in the presence of cyclostationarity, the available examples often correspond to finite time segments, extracted at different "start times" (relative to an arbitrarily chosen time instance), resulting in an *unsynchronized* dataset. In other words, additional latent variables, in the form of random time shifts, are introduced, making the problem more challenging.

[1] Note that the ability to synthetically generate a signal does not equate to having the capacity to specify its true statistics analytically.

This chapter gives particular attention to the scenario where each component is a randomly time-shifted and scaled segment[2] from a cyclostationary complex Gaussian (hereafter, simply referred to as *Gaussian* in this work) process. This is a regime in which an analytical form of the optimal estimator, in the sense of MMSE, can be expressed. In other words, the methods described herein aim to separate signals by exploiting the temporal structures of the latent components up to their second-order statistical characteristics.

Nonetheless, we also highlight the challenges in implementing the optimal estimator, thereby motivating the need for less computationally demanding alternatives. And while many signals of practical relevance deviate from the Gaussian model, this analysis involving cyclostationary Gaussian signals allows us to better understand and characterize the performance of our proposed data-driven methods. In particular, we focus on leveraging temporal correlation structures of the latent source components, and on assessing the ability of deep learning approaches to capture such structures. Through this framework, we can identify the performance gap and explore strategies to narrow it. Additionally, analyzing such a regime can provide insights from a model-based perspective. Finally, this characterization is accompanied by discussions on the appropriate neural architectures and parameterization choices for this problem.

Lastly, we end with a short discussion on the expected gains, in the sense of MSE, on different representative problem regimes. We also highlight cases where our optimism about performance improvements (based on second-order statistics alone) ought to be tempered.

## 4.1 Background on Cyclostaionary Signals

### 4.1.1 Cyclostationarity

Cyclostationary signals are characterized by statistical properties that vary periodically with time. Particularly, we are interested in signals that exhibit cyclostationarity in first and second-order statistics (corresponding to the mean and autocorrelation function)—also described as "wide-sense cyclostationary". Mathematically, a wide-sense discrete-time cyclo-

[2] The choice to account is a direct response to the limitations highlighted in the previous chapter, specifically the constraints regarding the relative time synchronization and location of the two sources.

stationary process, $x[n]$, with period $N_x$, has the following properties—

$$\mu_x[n] \triangleq \mathbb{E}\left[x[n]\right] = \mathbb{E}\left[x[n+N_x]\right], \forall n \in \mathbb{Z}, \tag{4.1}$$

$$C_x[n,l] \triangleq \mathbb{E}\left[x[n+l]\, x^*[n]\right] = \mathbb{E}\left[x[n+N_x+l]\, x[n+N_x]\right], \forall n,l \in \mathbb{Z}. \tag{4.2}$$

$\mu_x[n]$ and $C_x[n,l]$ correspond to the mean and autocorrelation function, respectively, and are terms that describe the random process $x[n]$. Note that a cyclostationary signal differs from a periodic signal in that its periodicity is in its statistical characteristics, and does not necessarily have regular repetitions of patterns in time.

### 4.1.2 Stationarizing and Ergodicity

Due to the nonstationary nature of cyclostationary random processes, consideration for the time index plays a vital role. For instance, consider a wide-sense cyclostationary random process with cyclostationary period $N_x$, described by the mean and autocorrelation function

$$\mu_{x1}[n] = \mu_{x1}[n+N_x]\ ,\ C_{x1}[n,l] = C_{x1}[n+N_x,l]$$

Let us introduce another random process where the statistics differ by $\tau$ time steps, i.e., with mean and autocorrelation function

$$\mu_{x2}[n] \triangleq \mu_{x1}[n+\tau]\ ,\ C_{x2}[n,l] \triangleq C_{x2}[n+\tau,l]$$

Note that these describe different processes, in that $\mu_{x1}[n], C_{x1}[n,l]$ and $\mu_{x2}[n], C_{x2}[n,l]$ are not necessarily equal (except in the case where $\tau$ is an integer multiple of $N_x$, by definition of cyclostationarity).

The observation about the distinction between processes with relative lag is an important one, particularly as we link it to another consideration in our problem. We make a key distinction between the random process versus the finite-length time series (also termed "sample paths" in some works [69]). When we describe a cyclostationary random process, it is defined for an infinite interval. However, in practice, we only observe a segment of finite length.

In practice, our dataset comprises finite-length records from this random process. Furthermore, we may not have explicit control over when this time series starts relative to the description of this time process. The lack of synchronization between these time series realizations introduces an additional source of randomness to their model.

One might be inclined to introduce the shifts $\tau$ as random variables into the description of the process. Hence, let us introduce a random process $\tilde{x}[n]$, related to the description of the cyclostationary process $x_1[n]$ with period $N_x$ by

$$\tilde{x}[n] = x_1[n+\tau]$$

for which $\tau$ is a random variable drawn from a uniform discrete distribution $\mathrm{unif}(0, N_x - 1)$. Inspecting the mean and covariance for this process

$$\begin{aligned}
\mu_{\tilde{x}}[n] &= \mathbb{E}\left[x_1[n+\tau]\right] \\
&= \frac{1}{N_x} \sum_{\tau=0}^{N_x-1} \mu_{x1}[n+\tau] \\
&= \frac{1}{N_x} \sum_{\tau=0}^{N_x-1} \mu_{x1}[\tau]
\end{aligned}$$

$$\begin{aligned}
C_{\tilde{x}}[n,l] &= \mathbb{E}\left[x[n+\tau+l]\, x^*[n+\tau]\right] \\
&= \frac{1}{N_x} \sum_{\tau=0}^{N_x-1} \mathbb{E}\left[x[n+\tau+l]\, x^*[n+\tau]\right] \\
&= \frac{1}{N_x} \sum_{\tau=0}^{N_x-1} C_{x1}[n+\tau, l] \\
&= \frac{1}{N_x} \sum_{\tau=0}^{N_x-1} C_{x_1}[\tau, l]
\end{aligned}$$

where for the last step, equality holds due to the cyclostationarity nature of $x_1$, meaning that any $N_x$ consecutive terms of $\mu_{x1}[n]$ sums to the same value, and similarly for any $N_x$ consecutive terms of $\boldsymbol{C}_{x1}[n,l]$. Note that, as a result of introducing the random shifts to the process, the mean and autocorrelation function of $\tilde{x}$ ends up being independent of time index $n$, i.e., that $\tilde{x}$ is a wide-sense stationary process. We remark that the properties of $\tilde{x}$, by introducing the random time shifts $\tau$, are indeed different from the random process we started with, namely $x_1$.

This has been an observation addressed in earlier works on cyclostationarity, under the lens of stationarizing the process or phase randomization [70]. Stationarizing refers to transforming a cyclostationary signal into a stationary one by averaging or other statistical operations. However, this process often results in the loss of important cyclostationary information, limiting the effectiveness of subsequent signal-processing algorithms.

The vital difference is that $\tilde{x}[n]$ is not an accurate model for time segments extracted from $x_1[n]$ with random starting point; for instance, when we choose a segment $[x_1[\tau], x_1[\tau+1], \ldots, x_1[\tau+N-1]$, the time offset $\tau$ is the same across these joint samples, even if it is random between different segments.

The analysis of an ensemble of such unsynchronized time series (sample paths) can be misleading as a result. In fact, we lose ergodicity, i.e., that taking the average over sample paths leads to empirical estimates of the stationarized process $\tilde{x}[n]$, rather than the true underlying cyclostationary process $x_1[n]$. The distinction of sample paths from the cyclostationary process with a random starting point and the stationarized process becomes critical in the context of our work. The following discussion in this chapter relates to the pitfall of "stationarizing" the model if we do not account for these time shifts in the examples in our dataset, and formulating the optimal estimators, which would be achieved only by considering the true underlying cyclostationarity of the signals.

## 4.2 Problem Formulation

We now formalize the particular signal separation problem for this chapter. Building upon the model established, we consider an observed signal of length $N$, which is a noisy mixture of two latent sources,

$$\boldsymbol{y} = \underbrace{\boldsymbol{s}_{\tau_s}}_{\text{signal-of-interest}} + \underbrace{\kappa\, \boldsymbol{b}_{\tau_b}}_{\text{interference}} + \underbrace{\sigma_w \boldsymbol{z}}_{\text{noise}} \in \mathbb{C}^N, \tag{4.3}$$

where $\boldsymbol{s}_{\tau_s}, \boldsymbol{b}_{\tau_b}$ are the unobserved independent components (with an additional subscript $\tau$ to explicitly model the time offsets, further elaborated later), $\kappa \in \mathbb{R}^+$ is the relative gain on the interference that is distributed according to some unknown (and for simplicity, discrete) distribution on $\mathcal{K} \subset \mathbb{R}^+$, and $\sigma_w \boldsymbol{z}$ is an additive white Gaussian noise, namely $\boldsymbol{z} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$, such that $\sigma_w^2$ corresponds to the variance of the additive noise component, where $\boldsymbol{0}$ denotes the $N$-length all-zeros vector and $\boldsymbol{I}$ denotes the $N \times N$ identity matrix.

Similarly, $\boldsymbol{s}_{\tau_s}$ is termed the "reference" signal (or also referred to as the SOI) and $\boldsymbol{b}_{\tau_b}$ the interference. Additionally, we assume that $\boldsymbol{s}_{\tau_s}$ and $\boldsymbol{b}_{\tau_b}$ have unit average power; hence, $\kappa$ is related to the inverse square root of the SIR. Recall that our performance metric is the MMSE of the estimate $\widehat{\boldsymbol{s}}_{\tau_s}$ from the observation $\boldsymbol{y}$.

We are particularly interested in the scenario where the signal models—i.e., the distributions of $\boldsymbol{s}_{\tau_s}$ and $\boldsymbol{b}_{\tau_b}$—are not explicitly known. Despite this, we operate under the assumption that we have access to a dataset of i.i.d. copies of $\{(\boldsymbol{y}^{(i)}, \boldsymbol{s}_{\tau_s}^{(i)})\}_{i=1}^{M}$, facilitating the use of a data-driven methodology.

We place particular focus on our discussion on signal components being segments extracted from cyclostationary Gaussian processes. We consider two independent, discrete-time, zero-mean circularly-symmetric Gaussian processes $\tilde{s}[\cdot]$, $\tilde{b}[\cdot]$, with autocovariance functions satisfying

$$\begin{aligned} C_{\tilde{s}}[n,l] &\triangleq \mathbb{E}\left[\tilde{s}[n+l]\,\tilde{s}^*[n]\right], \;\; C_{\tilde{s}}[n,l] = C_{\tilde{s}}[n+N_s,l], \\ C_{\tilde{b}}[n,l] &\triangleq \mathbb{E}\left[\tilde{b}[n+l]\,\tilde{b}^*[n]\right], \;\; C_{\tilde{b}}[n,l]= C_{\tilde{b}}[n+N_b,l], \end{aligned}$$

i.e., cyclostationary with fundamental periods $N_s, N_b > 1$ .

We denote the temporal offsets by $\tau_s$, $\tau_b$. Hence,

$$\boldsymbol{s}_{\tau_s} = [s_{\tau_s}[0], \ldots, s_{\tau_s}[N-1]]^{\mathrm{T}} \in \mathbb{C}^N, \; s_{\tau_s}[n] \triangleq \tilde{s}[n+\tau_s],$$

and similarly for $\boldsymbol{b}_{\tau_b}$ and $\tilde{\boldsymbol{b}}$. We consider the case where temporal offsets are random, drawn from a discrete uniform distribution, i.e., $\tau_s \sim \mathrm{unif}\{0, N_s-1\}$ and $\tau_b \sim \mathrm{unif}\{0, N_b-1\}$, and assume $\tau_s$, $\tau_b$, $\tilde{s}[\cdot]$, $\tilde{b}[\cdot]$ and $\kappa$ are statistically independent. Consequently, $\boldsymbol{s}_{\tau_s}$ and $\boldsymbol{b}_{\tau_b}$ are Gaussian mixtures with $N_s$ and $N_b$ components respectively.

Note that we consider our dataset to be made up of *unsynchronized* examples, wherein the corresponding time offsets $\tau_s$ and $\tau_b$ for each realization in the dataset are unknown.

## 4.3 Optimal Model-Based Estimators

We now derive, for several cases, optimal estimators that achieve the lower bounds for their respective cases. Specifically, these derivations, particularly the simplified expression of the MMSE estimator, not only reveal the challenges in realizing them, but also provides valuable intuition that informs and justifies our proposed data-driven approach to the signal separation problem.

One could consider a "classical signal processing" approach, typically involving a two-step process. Such an approach encompasses an initial estimation of the model parameters, and then subsequently an MMSE estimation based on the empirical model. However, estimation of these parameters requires synchronization of the given dataset, which can be a very challenging task in itself. Nevertheless, for the purposes of this section, we make the assumption of oracle access to the signal model, which enables us to establish a lower bound on the MSE for our problem.

Specifically, in the following, we assume oracle knowledge of the signal models—i.e., the first and second-order statistics—of $\tilde{\boldsymbol{s}}$ and $\tilde{\boldsymbol{b}}$, as well as the marginal distributions of $\tau_s, \tau_b$. We denote the conditional (temporal) covariance of the finite-length time series $\boldsymbol{s}_{\tau_s}$ given $\tau_s$ by

$$\boldsymbol{C}_{\boldsymbol{s}}(\tau_s) \triangleq \mathbb{E}\left[\boldsymbol{s}_{\tau_s}\boldsymbol{s}_{\tau_s}^{\mathrm{H}}|\tau_s\right] \in \mathbb{C}^{N\times N},$$

which is a function of $\tau_s$; likewise, $\boldsymbol{C}_{\boldsymbol{b}}(\tau_b)$ denotes the conditional covariance of $\boldsymbol{b}_{\tau_b}$. We denote the entries of $\boldsymbol{C}_{\boldsymbol{s}}(\tau_s)$ by $(\boldsymbol{C}_{\boldsymbol{s}}(\tau_s))_{i,j} = C_{\tilde{s}}[i+\tau_s, i-j]$.

### 4.3.1 Case 1: The Optimal Linear MMSE Solution

To obtain the second-order statistic of $\boldsymbol{s}_{\tau_s}$, one must account for the uniform randomness in the random variable $\tau_s$. Hence, the covariance of $\boldsymbol{s}_{\tau_s}$ is given by

$$\check{\boldsymbol{C}}_{\boldsymbol{s}} = \mathbb{E}\left[\boldsymbol{s}_{\tau_s}\boldsymbol{s}_{\tau_s}^{\mathrm{H}}\right] = \mathbb{E}_{\tau_s}\left[\mathbb{E}\left[\boldsymbol{s}_{\tau_s}\boldsymbol{s}_{\tau_s}^{\mathrm{H}}|\tau_s\right]\right] = \frac{1}{N_s}\sum_{\tau_s=0}^{N_s-1}\boldsymbol{C}_{\boldsymbol{s}}(\tau_s).$$

Note that this corresponds to a Toeplitz covariance structure, due to the fact that $s_{\tau_s}[\cdot]$ is a wide-sense stationary process, unlike $\tilde{s}[\cdot]$ (as elaborated earlier in Section 4.1.2). The same applies to the covariance of $\boldsymbol{b}_{\tau_b}$.

The second-order statistics of $\boldsymbol{s}_{\tau_s}$ and $\boldsymbol{b}_{\tau_b}$ are sufficient for optimal *linear* estimation. For the sake of practicality, if we restrict our attention to linear operators, the optimal estimator is given by

$$\widehat{\boldsymbol{s}}_{\tau_s,\mathrm{LMMSE}} = \check{\boldsymbol{C}}_{\boldsymbol{s}} \left[\check{\boldsymbol{C}}_{\boldsymbol{s}} + \mathbb{E}\left[\kappa^2\right] \cdot \check{\boldsymbol{C}}_{\boldsymbol{b}} + \sigma^2 \boldsymbol{I}\right]^{-1} \boldsymbol{y}, \tag{4.4}$$

where we used the statistical independence assumption of the sources and noise to compute the covariance of $\boldsymbol{y}$. Indeed, this would achieve the MMSE within the family of linear estimators. Further, given a sufficiently large dataset, it is feasible to implement an accurate approximation of (4.4) by substituting the covariance matrices with their empirical estimates (i.e., sample covariances) derived from the datasets. However, due to the random time offsets, $\boldsymbol{s}_{\tau_s}$ and $\boldsymbol{b}_{\tau_b}$ are not Gaussian—but instead, are Gaussian mixtures. Therefore, the optimal linear estimator does not coincide with the MMSE estimator.

### 4.3.2 Case 2: The Oracle MMSE Solution

In this subsection, we develop the optimal solution for the case where $(\tau_s, \tau_b, \kappa)$ are known. This would be the case if we had access to an oracle providing side information for perfect synchronization to both the reference and interference signals, as well as the SIR-related coefficient $\kappa$. Under this setting, $\boldsymbol{y}$ and $\boldsymbol{s}_{\tau_s}$ are jointly Gaussian; therefore, the optimal solution can be given by

$$\widehat{\boldsymbol{s}}_{\tau_s}|_{(\tau_s,\tau_b,\kappa)} = \mathbb{E}\left[\boldsymbol{s}_{\tau_s}|\boldsymbol{y}, \tau_s, \tau_b, \kappa\right] = \boldsymbol{H}(\tau_s, \tau_b, \kappa)\boldsymbol{y}, \tag{4.5}$$

where

$$\boldsymbol{H}(\tau_s, \tau_b, \kappa) \triangleq \boldsymbol{C}_{\boldsymbol{s}}(\tau_s)\left[\boldsymbol{C}_{\boldsymbol{s}}(\tau_s) + \kappa^2 \boldsymbol{C}_{\boldsymbol{b}}(\tau_b) + \sigma^2 \boldsymbol{I}\right]^{-1} \tag{4.6}$$

is the optimal linear time-varying filter. Since (4.6) is a function of $(\tau_s, \tau_b, \kappa)$, we subsequently refer to (4.5) as an oracle-synchronized MMSE solution. Nevertheless, it is important to highlight that (4.5) is not a realizable estimator, as it is a function of latent, unobservable variables. We also remark that, for cyclostationary signals, (4.5) can also be represented as a frequency-shift filter or a cyclic Wiener filter, i.e., expressed as a linear function of frequency-shifted copies of the observation [9, 67].

It should be noted that every $\tau_s$, $\tau_b$ and $\kappa$ configuration leads to a corresponding optimal linear filter. As we consider various relative shifts and SIR levels, the number of linear filters grows as the product of these corresponding degrees of freedom.

### 4.3.3 Case 3: The Optimal MMSE Estimator

In general, the time offsets $\tau_s$ and $\tau_b$, and the SIR parameter $\kappa$, are not known at inference time. A realizable estimator must therefore account for their inherent randomness either by explicit or implicit (and generally non-linear) estimation. Nonetheless, upon conditioning on these quantities, i.e., considering (4.6) as fixed, the resulting optimal MMSE estimator would be linear. With this observation in mind, the true, realizable MMSE estimator can be expressed as

$$\begin{aligned}\widehat{\boldsymbol{s}}_{\tau_s,\text{MMSE}} &= \mathbb{E}\left[\boldsymbol{s}_{\tau_s}|\boldsymbol{y}\right] = \mathbb{E}_{(\tau_s,\tau_b,\kappa)|\boldsymbol{y}}\left[\mathbb{E}\left[\boldsymbol{s}_{\tau_s}|\boldsymbol{y},\tau_s,\tau_b,\kappa\right]\right]\\ &= \sum_{\tau_s=0}^{N_s-1}\sum_{\tau_b=0}^{N_b-1}\sum_{\kappa\in\mathcal{K}} p(\tau_s,\tau_b,\kappa|\boldsymbol{y})\,\widehat{\boldsymbol{s}}_{\tau_s}|_{(\tau_s,\tau_b,\kappa)},\end{aligned} \tag{4.7}$$

which essentially corresponds to the sum of oracle-synchronized MMSE solutions (4.5) for each set of parameters, weighted by the corresponding posterior probabilities given the observation $\boldsymbol{y}$. We emphasize that (4.7), unlike (4.5), is a legitimate estimator—that is, a function of only the observed data. However, the MSE achieved by (4.5) serves as a lower bound of that by (4.7), by data processing inequality of MMSE.

It should be noted that, in cases when the true posterior is a Kronecker delta function, (4.7) in fact coincides with (4.5),

$$\begin{aligned}& p(\tau_s,\tau_b,\kappa|\boldsymbol{y}) = \delta[\tau_s-\tau_s^*,\tau_b-\tau_b^*,\kappa-\kappa^*]\\ \Longrightarrow\quad & \widehat{\boldsymbol{s}}_{\tau_s,\text{MMSE}} = \widehat{\boldsymbol{s}}_{\tau_s}|_{(\tau_s^*,\tau_b^*,\kappa^*)}.\end{aligned} \tag{4.8}$$

This suggests that, under such conditions, the performance of the oracle-aided solution is attainable.

## 4.4 Methodologies for Data-Driven Signal Separation

The methods discussed thus far provide an understanding of the functional structure of the MMSE estimator, in terms of the (unknown) time offsets and SIR parameters. Nevertheless, implementing the MMSE estimator may not be possible in practice, due to the challenges outlined below.

### 4.4.1 Difficulties in Realizing the MMSE Estimator

First, as seen in (4.7), the MMSE estimator involves the computation of a posterior term over the latent variables $(\tau_s, \tau_b, \kappa)$. However, obtaining this posterior becomes computationally intensive as the space of parameters increases.

Second, (4.7) also involves a sum of different (conditionally) linear operators, each corresponding to a set of parameter values. However, each of these operators involves an inversion of a large covariance matrix, which is infeasible in regimes of signals from long observation periods (large $N$).

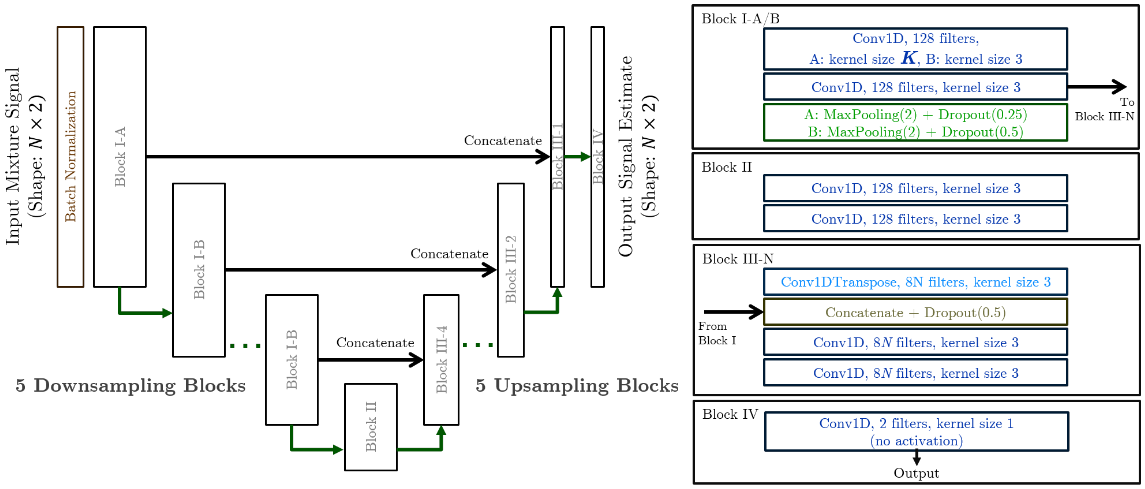
Lastly, and importantly, in practice, we do not have oracle access to the signal model, corresponding to the first and second-order statistics in the context of the Gaussian models. Instead, we are provided with many examples, through which we can obtain empirical estimates of the statistics. We reemphasize that the dataset of the signals is not synchronized, meaning that to obtain the second-order statistics of the underlying cyclostationary signal, $\boldsymbol{C}_{\tilde{\boldsymbol{s}}}$ and $\boldsymbol{C}_{\tilde{\boldsymbol{b}}}$, we need to estimate the latent variable $\tau_s$ and $\tau_b$, for each example in the dataset. This corresponds to synchronizing the entire dataset, which is generally a challenging task of independent interest in itself. This last issue presents difficulties when implementing both the optimal linear filters and the computation of the posterior term, given the lack of knowledge about the synchronized covariance matrices.

We now show that machine learning methods, specifically deep neural networks, can successfully navigate the aforementioned challenges. These neural networks can be trained on *unsynchronized* datasets to solve the separation problem. We present two representative examples to demonstrate our approach, and compare it with the performance of an optimal MMSE solution that leverages oracle knowledge. Our primary objective henceforth is to establish a practical pipeline, and benchmark it against a theoretical lower bound.

### 4.4.2 Supervised Separation with U-Net

Given the formulation in (4.3), a natural approach is to use a deep neural network to learn a regression model with multivariate output for source separation. We propose to use the so-called "U-Net" architecture for signal separation (architecture shown in Fig. 4-1).[3] Such a neural network architecture was first proposed for biomedical image segmentation [54],

[3] https://github.com/RFChallenge/SCSS_CSGaussian


Input Mixture Signal
(Shape: $N \times 2$)
Batch Normalization
Block I-A
Block I-B
Block I-B
Block II
Concatenate
Concatenate
Concatenate
Block III-4
Block III-2
Block III-1
Block IV
Output Signal Estimate
(Shape: $N \times 2$)
5 Downsampling Blocks
5 Upsampling Blocks
Block I-A/B
Conv1D, 128 filters,
A: kernel size $K$, B: kernel size 3
Conv1D, 128 filters, kernel size 3
A: MaxPooling(2) + Dropout(0.25)
B: MaxPooling(2) + Dropout(0.5)
To
Block III-N
Block II
Conv1D, 128 filters, kernel size 3
Conv1D, 128 filters, kernel size 3
Block III-N
Conv1DTranspose, 8N filters, kernel size 3
From
Block I
Concatenate + Dropout(0.5)
Conv1D, $8N$ filters, kernel size 3
Conv1D, $8N$ filters, kernel size 3
Block IV
Conv1D, 2 filters, kernel size 1
(no activation)
Output

All Conv1Ds and Conv1DTransposes use ReLU activation

Figure 4-1: The U-Net architecture used for source separation in our simulations.

but has found its use in other applications, including spectrogram-based RF interference cancellation [8] and audio source separation [23, 32]—all of which also corresponding to a multivariate regression setup with the same dimensions on the input and output. Similar to the latter works, we use 1D-convolutional layers to capture features relating to the time series data. The U-Net architecture contains downsampling blocks that operate on successively coarser timescales and possesses skip connections that combine features at these various timescales with the upsampling blocks.

To process complex-valued signals, borrowing inspiration from widely linear estimation [71], we stack the real and imaginary parts as separate channels to the U-Net.

As these methods are applied to time series signals in practice, using domain knowledge to craft an appropriate neural network architecture may be crucial in attaining performance gain, as evident in our experiments and architectural choices. For example, we made the intentional choice of longer kernel sizes on the first convolutional layers. This further reinforces the relevance of this work, that is, in identifying and characterizing neural network architectures under study relative to the best possible performance. In our experiments, we observe that kernel sizes that match the effective correlation length (i.e., timescales in which the covariance magnitudes are non-negligible) are required to attain the best performance. This may be an indication to how some partial, though important, information about the signal model is helpful (or even essential) in seeking the appropriate neural network architecture.

## 4.5 Computational Simulation

We now consider two examples for signal separation. For each setting, we train a U-Net to estimate the corresponding signal $\boldsymbol{s}_{\tau_s}$ from the mixture $\boldsymbol{y}$ in an end-to-end fashion, and with no supervision regarding the time-shifts $\tau_s$ and $\tau_b$, and the gain $\kappa$.

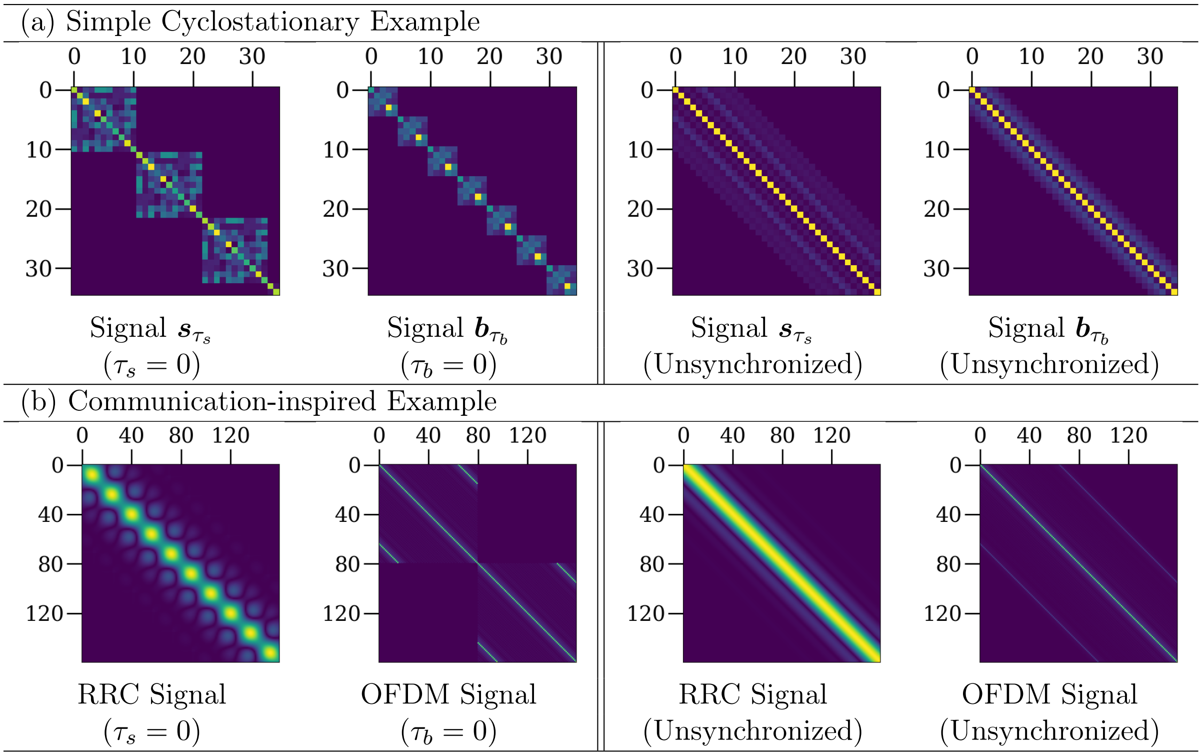
In the examples below, we describe how long (relative to the window length $N$, but still finite) segments of the processes $\tilde{s}[\cdot], \tilde{b}[\cdot]$ are generated, from which $N$-length segments are extracted to create the datasets. The training set is processed as such to yield a labeled dataset of i.i.d. copies—mixture and ground-truth reference signal, $\{(\boldsymbol{y}^{(i)}, \boldsymbol{s}_{\tau_s}^{(i)})\}_{i=1}^{M}$—as is done in the supervised learning framework. Our training set comprises $10,000 \times |\mathcal{K}|$ pairs of mixtures $\boldsymbol{y}$ and ground-truth $\boldsymbol{s}_{\tau_s}$, and the validation set comprises $500 \times |\mathcal{K}|$ pairs, where the cardinality $|\mathcal{K}|$ is the total number of levels for $\kappa$ under consideration. Subsequently, we test the performance across $1,000$ examples per $\kappa$ level, reporting the average MSE in dB. Note that varying $\kappa$ results in different levels of SIR. In our simulations, we assess the separation performance across different SIR levels. We also compare the performance of using the aforementioned U-Net against that of the optimal model-based estimators. For these optimal estimators, $\kappa$ is assumed to be known.

**Implementation Details:** Keras and Tensorflow 2 are used to implement and train the U-Net models [72, 73]. For training, we use empirical MSE as the loss function. We also use Adam optimizer [74] and an exponentially decaying learning rate schedule, batch size of 32 with shuffled training samples, and trained for 2,000 epochs with early stopping if there is no improvement for 100 epochs on the validation set. We train the neural networks on a computing node from a high-performance computing cluster with Intel Xeon Gold 6248, 192 GB RAM, and an NVidia Volta V100 GPU.

### 4.5.1 Signals from Randomly Generated Covariances

We consider (1.1) with $N = 256$, $N_s = 11$, $N_b = 5$, $\mathcal{K}$ corresponding to 5 equidistant SIR levels in $[-6, 6]$ dB. The reference and interference signals are generated as

$$\tilde{\boldsymbol{s}} = \boldsymbol{G_s a}_1, \ \ \tilde{\boldsymbol{b}} = \boldsymbol{G_b a}_2,$$


(a) Simple Cyclostationary Example

Signal $\boldsymbol{s}_{\tau_s}$ ($\tau_s = 0$)

Signal $\boldsymbol{b}_{\tau_b}$ ($\tau_b = 0$)

Signal $\boldsymbol{s}_{\tau_s}$ (Unsynchronized)

Signal $\boldsymbol{b}_{\tau_b}$ (Unsynchronized)

(b) Communication-inspired Example

RRC Signal ($\tau_s = 0$)

OFDM Signal ($\tau_b = 0$)

RRC Signal (Unsynchronized)

OFDM Signal (Unsynchronized)

Figure 4-2: Visualization of (a sub-matrix of) the covariance matrices for the signals used in the respective examples.

where $\boldsymbol{a}_1, \boldsymbol{a}_2 \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$, and $\boldsymbol{G_s}, \boldsymbol{G_b} \in \mathbb{C}^{\tilde{N} \times \tilde{N}}$, with $\tilde{N} = 550$, are block-diagonal matrices with repeating $N_s \times N_s$ and $N_b \times N_b$ blocks respectively. Each entry in the blocks is drawn once independently from the Gaussian distribution, and is fixed for the rest of this experiment. Full details on the signal generation are provided in our Github repository.[3] Fig. 4-2(a) shows the covariance structures of the resulting sources.

Fig. 4-3 compares the MSE achieved by the U-Net against that obtained by the linear and the "global" MMSE estimators. We also include the oracle-synchronized MMSE (4.5), which, as evident from the figure, is indistinguishable, in terms of its MSE, from the true, non-oracle MMSE estimator (4.7). This occurs when all the mass of the posterior is approximately concentrated at the point of the true values of $(\tau_s, \tau_b, \kappa)$, as in (4.8).

We observe that a U-Net trained on an unsynchronized dataset of signals is capable of obtaining results close to the MMSE performance. This means that the U-Net necessarily learned a significant part of the model, which enables high-quality estimation of the SOI. We reiterate that the U-Net did not have access to the true statistics of the signal model or any form of explicit synchronization of the signals during training and inference. The slight

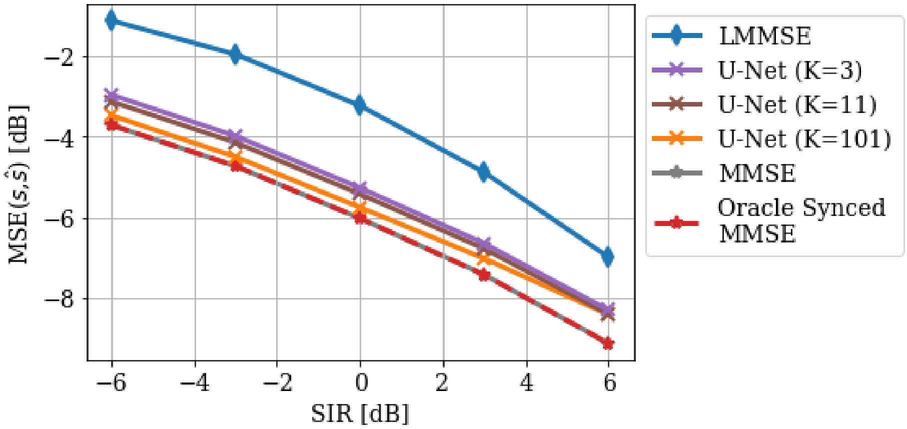


Figure 4-3: Separation performance of the U-Net separator vs. optimal model-based estimators for waveforms with randomly generated covariance structures.

deterioration performance could be attributed to approximation errors introduced from a deep learning-based function approximator, or due to the trade-off from lack of access to a synchronized dataset. Future work entails identifying factors to close this performance gap.

### 4.5.2 Communication-like Waveforms

In this example, we consider two types of signals with second-order statistical properties resembling a single-carrier communication waveform and an OFDM waveform, respectively. The single-carrier signal is modeled as,

$$\tilde{s}[n] = \sum_{p=-\infty}^{\infty} a_p \, g[n - pN_s], \tag{4.9}$$

where $a_p \sim \mathcal{CN}(0,1)$, $N_s$ is the symbol period, and $g[n]$ is the RRC filter that corresponds to 16 samples per symbol—i.e., $N_s = 16$—and spans 8 symbols, with a roll-off factor of 0.5. This also corresponds to an example where a cyclostationary signal's covariance (Fig. 4-2(b)) is not block-diagonal, which leads to an additional computational burden in the matrix inversion.

The second source, an OFDM waveform, is modeled as,

$$\tilde{b}[n] = \frac{1}{\sqrt{N_{\mathrm{sc}}}} \sum_{p=-\infty}^{\infty} \sum_{\ell=0}^{N_{\mathrm{sc}}-1} a_{p,\ell} \, q[n - pN_b - N_{\mathrm{cp}}, \, \ell], \tag{4.10}$$
$$q[n,\ell] = \mathbb{1}_{\{N_{\mathrm{cp}} \le n \le N_{\mathrm{sc}}-1\}} \cdot \exp\left(j2\pi\ell \frac{n}{N_{\mathrm{sc}}}\right),$$

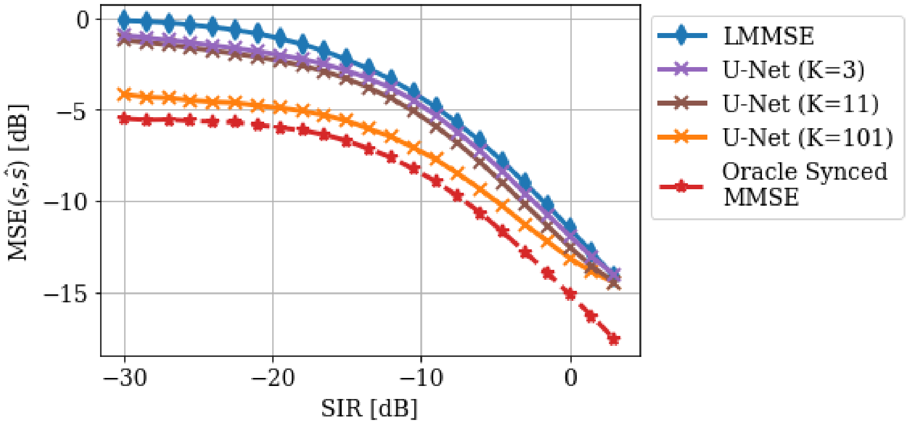


Figure 4-4: Separation performance of the U-Net separator vs. optimal model-based estimators for communication-like waveforms.

where $a_{p,\ell} \sim \mathcal{CN}(0,1)$ for $\ell \in \mathcal{L}_{\text{sc}}$, where $\mathcal{L}_{\text{sc}}$ refers to the set of nonzero subcarrier indices, and $a_{p,\ell} = 0$ otherwise, $N_{\text{sc}}$ is the total number of subcarriers (both nonzero and null ones) per OFDM symbol, $N_{\text{cp}}$ is the cyclic prefix (CP) length, and $N_b$ is the OFDM symbol period where $N_b = N_{\text{sc}} + N_{\text{cp}}$. In our specific example, we chose $N_{\text{sc}} = 64$, $N_{\text{cp}} = 16$, and thus $N_b = 80$; this is loosely based on parameters from 802.11n WiFi waveform properties [62]. More details on the signal specifications are provided in our Github repository.[3] Fig. 4-2(b) shows the covariance structures of the resulting sources, whose cyclostationarity is evident.

We consider segments of length $N = 1280$, and $\mathcal{K}$ corresponding to SIR levels from $-30$ dB to 3 dB at 1.5 dB steps ($|\mathcal{K}| = 23$), focusing on the more challenging low SIR regime. We emphasize that the models (4.9) and (4.10) are assumed to be *unknown* once we have generated the dataset. Rather, we only have access to a dataset of unsynchronized samples.

Fig. 4-4 compares the MSE achieved by the linear MMSE and the oracle-synchronized MMSE. We note that, for this example, the true MMSE curve could not be obtained in practice due to the size of the parameter space, rendering the computation of the posterior infeasible. Nevertheless, the MSE of the oracle-synchronized MMSE solution—albeit not a realizable estimator, as established earlier—serves as a lower bound.

As observed from Fig. 4-4, the best performing U-Net, which is trained on unsynchronized data, outperforms the linear MMSE estimator, and is close to the performance of the oracle MMSE (e.g., about 1.2 dB away at SIR levels between $-9$ and $-30$ dB). We also highlight that the choice of U-Net architecture to achieve such performance benefits from specific domain knowledge. For example, capturing the temporal structures on the order of the signals' effective correlation length yielded significantly improved performance—for which long kernels on the first layer is one way of doing so.[4]

## 4.6 Residual Error Statistics

Building further on the results for the Communication example, we look at the residual errors in time. Note that based on (4.5), we can compute the MMSE at each time index, conditioning on knowing the true time offsets and SIR; the MMSE is expressed as

$$\boldsymbol{C_e}(\tau_s, \tau_b, \kappa) \triangleq \boldsymbol{C_s}(\tau_s) - \boldsymbol{C_s}(\tau_s)\left[\boldsymbol{C_s}(\tau_s) + \kappa^2 \boldsymbol{C_b}(\tau_b) + \sigma^2 \boldsymbol{I}\right]^{-1} \boldsymbol{C_s}^{\mathrm{H}}(\tau_s) \tag{4.11}$$

$$\mathrm{MMSE}(s_{\tau_s}[n], \widehat{s}_{\tau_s}[n]; \tau_s, \tau_b, \kappa) = \mathrm{Diag}(\boldsymbol{C_e}(\tau_s, \tau_b, \kappa))[n] \tag{4.12}$$

$$\text{TA-MMSE}(\tau_s, \tau_b, \kappa) = \frac{1}{N}\mathrm{Tr}(\boldsymbol{C_e}(\tau_s, \tau_b, \kappa)) \tag{4.13}$$

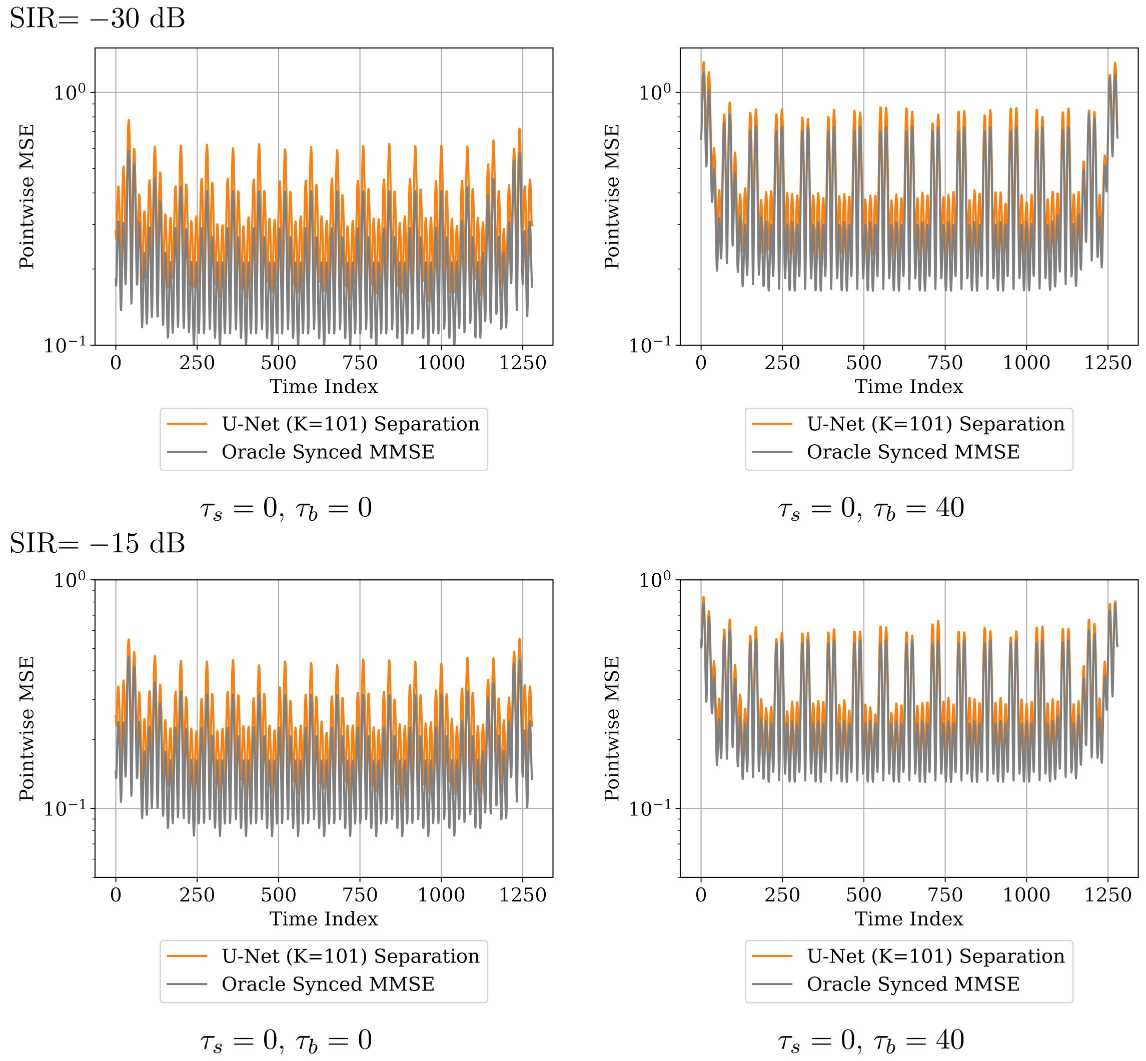
where $\mathrm{Diag}(\cdot)$ refers to taking the diagonal of the matrix, and $\mathrm{Diag}(C_e)[n]$ means to take the $n$-th element on the aforementioned diagonal. Note that we discussed the time-averaged MMSE, which in this case, would be obtained by taking the trace of the error covariance.

Fig. 4-5 shows the MMSE across the time indices for a few representative configurations of time offsets and SIR. Notably, beyond similarity in the time-averaged MSE sense as discussed earlier, the profile of the pointwise MMSE in time is also similar, reflecting how the trained U-Net is behaving as a good approximate to the oracle-synced solution (and thereby, potentially the actual MMSE estimator, which we are unable to compute feasibly).

## 4.7 Expected MSE for representative cases

As a detour, we explore the advantages of modeling the signals as time-shifted segments from cyclostationary Gaussian processes, namely in benchmarking and analysis with such an analytically tractable model. This approach provides valuable insights into potential performance gains by leveraging the joint statistics of the SOI and interference. Previous results have demonstrated that the MMSE estimator outperforms methods that naively assume temporal stationarity on the signals, such as using the LMMSE estimator. By establishing a benchmark for achievable performance, we can design a machine learning

[4] For comparisons with other deep neural networks for the communication example: https://github.com/RFChallenge/SCSS_DNN_Comparison



SIR$= -30$ dB

$\tau_s = 0,\ \tau_b = 0$

$\tau_s = 0,\ \tau_b = 40$

SIR$= -15$ dB

$\tau_s = 0,\ \tau_b = 0$

$\tau_s = 0,\ \tau_b = 40$

Figure 4-5: MSE across the time indices, as computed by (4.13), for four representative configurationss

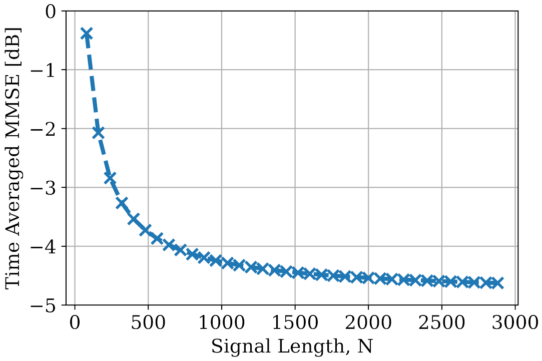


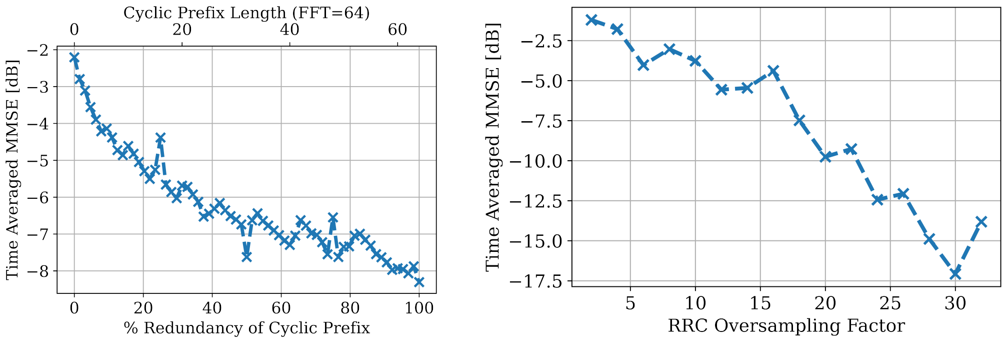
Figure 4-6: Expected MSE for different signal lengths, plotting for configuration at −30 dB SINR and for $\tau_s = 0$ and $\tau_b = 40$.

architecture that strives to approach this lower bound and exploits the temporal structures that capture the underlying cyclostationarity/second-order statistical structures.

In this section, we characterize a few variations in the signal specifications to show how the expected MMSE changes accordingly. Particularly, we examine the oracle synchronized cases, where we select representative values of time shifts and SIR levels; in such cases, the optimal solution corresponds to a linear estimator, as in (4.5). By analyzing these cases, we aim to glean insights into how changes in signal parameters may impact the achievable MMSE (or at least a lower bound on it), and ultimately inform our understanding of the problem at hand. We remark that while our discussion in this section centers on a single-carrier SOI with an OFDM interference (featuring periodic extension structures in the form of cyclic prefixes), the methodology described herein can be generalized to any cyclostationary Gaussian time series.

First, we examine the influence of signal lengths on the separation performance. A longer signal length allows for capturing longer temporal correlations. As a result, the algorithm can exploit these extended dependencies to achieve better performance. To illustrate this relationship between signal length and estimation accuracy, Fig. 4-6 shows how the time-averaged MMSE decreases as the signal length increases. This empirical observation highlights the importance of longer signal lengths to leverage the underlying structures better.

Another important factor to consider is the temporal characteristics of the signal components. In communication signals, there are specific properties that can be leveraged to improve the performance of source separation algorithms. These properties arise from the



(a) (b)

Figure 4-7: Expected MMSE for different configurations, corresponding to (a) different cyclic prefix lengths on the OFDM interference (with FFT length 64), leading to different degrees of redundancy per OFDM symbol; and (b) different oversampling factors on the root-raised cosine SOI, leading to different degrees of redundancy per SOI symbol.

departure of the signals from being i.i.d. or temporally stationary with Gaussian statistics. Two such properties that influence the signals' joint statistics are the oversampling rate of the RRC filter, which corresponds to the pulse width of the correlations, and the presence of a cyclic prefix in OFDM, which provides a level of redundancy.

Fig. 4-7 illustrates the influence of these temporal characteristics impacting the signal estimation performance. We demonstrate how varying the oversampling rate and the cyclic prefix length affect the MMSE (in the oracle-synchronized setting). [5] It is worth noting that the degree to which the signals can be separated depends on the joint statistics of the components present. Fig. 4-7 hence serves as a useful visual aid that illustrates the relationship between the temporal characteristics of the components and the resulting signal estimation performance. In turn, this may provide insights for scenarios where selecting specific signal parameters for better signal separation is desirable, though such considerations lie beyond the scope of this work.

Lastly, we also consider the presence of underlying AWGN in the dataset. In many practical scenarios, the observed signals are contaminated by noise, and this noise can significantly impact the data-driven algorithm. Specifically, the interference component/dataset may inherently contain noise, and the scaling factor $\kappa$ would also scale and amplify the noise, along with the interference waveform. In other words, revisiting (4.3), we consider the scenario where the interference dataset we have access to is noisy, resulting in the following

[5] It should be noted that the parameters are dictated by the properties of sources present, and are generally not tunable at the receiver. The goal hence is to assess what is the expected best performance by optimally exploiting their joint statistics.

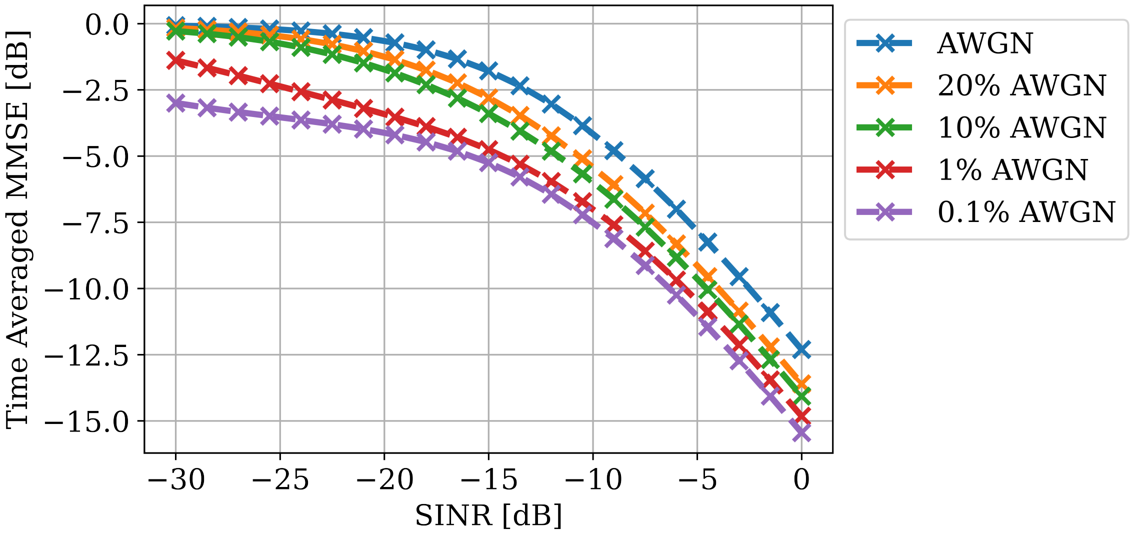


Figure 4-8: Expected MSE for when interference contains an AWGN component which is magnified by the gain $\kappa$.

representation

$$\boldsymbol{y} = \underbrace{\boldsymbol{s}_{\mathcal{T}_s}}_{\text{SOI}} \quad + \quad \kappa \quad \underbrace{(\boldsymbol{b}_{\mathcal{T}_b} + \boldsymbol{w})}_{\text{interference + noise}} \quad . \tag{4.14}$$

We investigate how the presence of noise in the interference dataset affects the performance of separating and estimating the SOI.

Fig. 4-8 depicts the expected impact of different AWGN levels modeled within the interference on the MMSE. As the level of AWGN in the interference component becomes stronger, the estimation performance is adversely affected, leading to increased MMSE. It is worth noting that when the noise is around 10% of the power relative to the total interference-plus-noise term, the MMSE attainable at the lower SINR levels is not significantly better than naively assumed stationarity (LMMSE) or if naively treated as AWGN. This observation underscores the challenges faced in real-world scenarios with low SNR datasets.

## 4.8 Going beyond Gaussianity

Reviewing the results presented thus far, we recognize that the Gaussianity assumption is made mainly for mathematical convenience, but many RF signals have non-Gaussian characteristics. This motivates the next part of the exploration, where we assess the performance of our proposed learning-based methods on digital communication signals.

In digital communication, we are primarily interested in extracting the encoded information bits that the SOI $\boldsymbol{s}_{\mathcal{T}_s}$ carries, rather than the quality of reconstructing the raw waveform. The key objective is to mitigate interference, facilitate a more accurate demodulation and recovery of the bits in the SOI. Still, signal separation remains an essential signal processing tool, especially in extracting the SOI component. Once we extract the SOI, standard tools and processing pipelines can be applied to this estimate of the SOI, presumably of higher fidelity. For example, after signal separation, we can apply conventional demodulation procedures to the estimate $\widehat{\boldsymbol{s}}_{\mathcal{T}_s}$ (e.g., matched filtering, by treating the residual errors as noise) to recover the information bits.

To illustrate this, we revert to the setup involving a root-raised cosine signal and an OFDM interference. Now, instead of Gaussian statistics, we consider waveforms that carry digital data, i.e., that their coefficients are drawn from a discrete coefficient set (e.g., a QPSK constellation). In other words, these non-Gaussian signals have higher-order statistical structures. By training the neural network on these signals, we can achieve better MSE performance and even outperform the theoretical baseline for the Gaussian case—which no longer serves as the lower bound in this non-Gaussian context—(Fig. 4-9).

To assess the resulting fidelity of the signal extracted, we can measure the bit error rate (BER) of the demodulated single-carrier waveform. By using the neural network separator as a pre-processing step, we can indeed achieve a lower BER compared to standard linear processing (matched filtering and LMMSE estimation), as demonstrated in Fig. 4-9. Further, training a neural network on non-Gaussian communication waveforms outperforms that of training solely on Gaussian waveforms, reflecting the former's ability to exploit structures beyond second-order statistical structures.

In the next section, we delve deeper into the example of separating an RRC QPSK SOI from an OFDM interference, and further explore other aspects of training such a deep learning architecture for the signal separation problem.

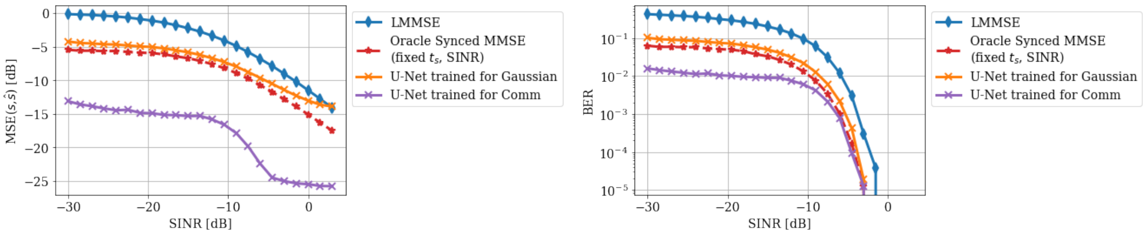


Figure 4-9: Comparison of MSE and BER in the single-channel source separation of digital communication signals. In this example, we are interested in the signal separation/interference mitigation of a single-carrier QPSK signal in the presence of an OFDM interference (with QPSK subcarrier symbols).

# Chapter 5

# Parameters and Hyperparameters for Deep Learning Architectures

Deep learning models are highly flexible and adaptable, allowing for a wide range of parameterization choices. The selection of appropriate hyperparameters, such as learning rate and batch size, also plays a crucial role in the overall performance of the models. These factors significantly influence the model's learning dynamics and convergence behavior.

The goal of this section is to uncover the intricacies pertaining to the engineering processes and hyperparameter characterization for the deep learning approach. In particular, we focus on the U-Net architecture studied in the previous chapter, and characterize some of the hyperparameter choices, leading to the best empirical result for the problem at hand.

This chapter, however, does not aim to be an exhaustive or definitive prescription of hyperparameters for the signal separation problem. Rather, this investigation serves as a useful window into understanding the effects and importance of various facets of the training process. The insights gleaned from this discussion would help in guiding future improvements and optimizations of the U-Net-based separation method.

## 5.1 Recap: Specifications of Waveforms under Study

We review the formulation for the source separation problem with RF signals, where we consider the following model for the observed mixture signal,

$$\boldsymbol{y} = \boldsymbol{s} + \kappa \boldsymbol{b},$$

where $\boldsymbol{s}$ and $\boldsymbol{b}$ correspond to the (unobserved) SOI and the interference components respectively, and $\kappa$ corresponds to the relative gain between the two components. In this chapter, we consider the SOI to be a single-carrier QPSK signal modulated by the RRC pulse shaping function, corresponding to a 16× oversampling factor; the interference is OFDM with NFFT $= 64$, cyclic prefix $N_{\mathrm{cp}} = 16$, and QPSK symbols populating 56 subcarriers per OFDM symbol. We focus on window length of $N = 2560$, and SIR levels from $-30$ dB to 3 dB.

Recall that we have no explicit knowledge about the parameters outlined above; instead, we have access to sample realizations of the signals, and have to learn their characteristics through data.

This case serves as a representative example of a scenario that we have found to be empirically challenging. It underscores the fact that substantial performance improvements over conventional linear processing can only be achieved through carefully chosen neural architectures, as detailed in the earlier chapters. Therefore, investigations using this example reflect the significance of the appropriate parameterization and hyperparameter selection in training deep learning models and the complexities associated with their optimal configuration.

### 5.1.1 Implementation Details

Keras and Tensorflow 2 are used to implement and train the neural network models mentioned in this chapter [72, 73]. For training, we use empirical MSE as the loss function. We train the neural networks on a computing cluster with Intel Xeon Gold 6248, 192 GB RAM, and an NVidia Volta V100 GPU.

As the default configuration for training parameters, we use a batch size of 256, a training set size of 250,000, and SIR levels drawn uniformly on the dB scale (i.e., logarithmic scale) between $-33$ dB and 3 dB, over 2000 epochs. Changes to these parameters are addressed in the subsequent subsection. A validation set is created using 1000 examples per SIR level, over a range of $-30$ dB to 3 dB in 1.5 dB steps. For all implementations, we used Adam optimizer [74] with a fixed learning rate of 0.0003 for a controlled comparison. We note that other adaptive learning rate schedules and strategies could potentially further improve results, and we leave this for further model optimization in future investigations.

Tensorboard is used to track the training and validation loss over the course of training.

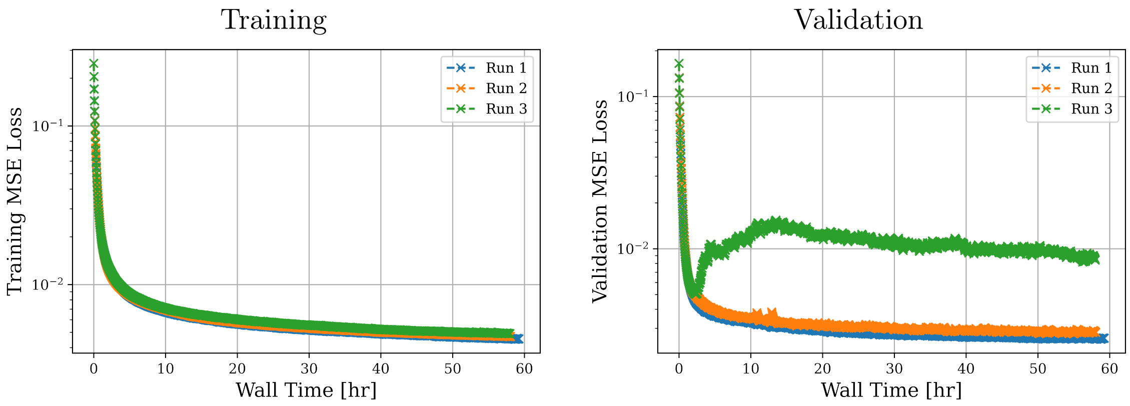


Figure 5-1: Variability in the training and validation losses between independent runs when training the Separation U-Net.

The losses reported in these performance traces are the mean MSE loss across all the SIR levels over the respective datasets.

### 5.1.2 Hyperparameters for Training

When training the neural networks, repetition of experiments under the same conditions might yield slightly different results, even when using the same random seed. This **variability** is primarily due to the inherent nondeterministic behaviors of the parallel computation with GPUs, leading to different final results.

It is essential to acknowledge this inherent variability. Therefore, if we focus on a single configuration, we can conduct multiple runs and select the best model based on the validation performance. This approach yields a more robust model and mitigates the risk of overfitting on a particular training run. Nevertheless, it should be noted that as the number of different configurations (e.g., mixtures of different signal types) grows, conducting repeated runs for each configuration can quickly become resource-intensive.

In our experiments, we look into the training and validation variability across three independent runs. The performance traces across time are presented in Fig. 5-1, reflecting variability on the order of $10^{-2}$. Although seemingly small, these differences may be significant in contexts where precision is critical. This underscores the need for caution when interpreting results, especially when determining statistical significance related to performance improvements.

One crucial aspect of training deep learning models is the **training set size**. A small

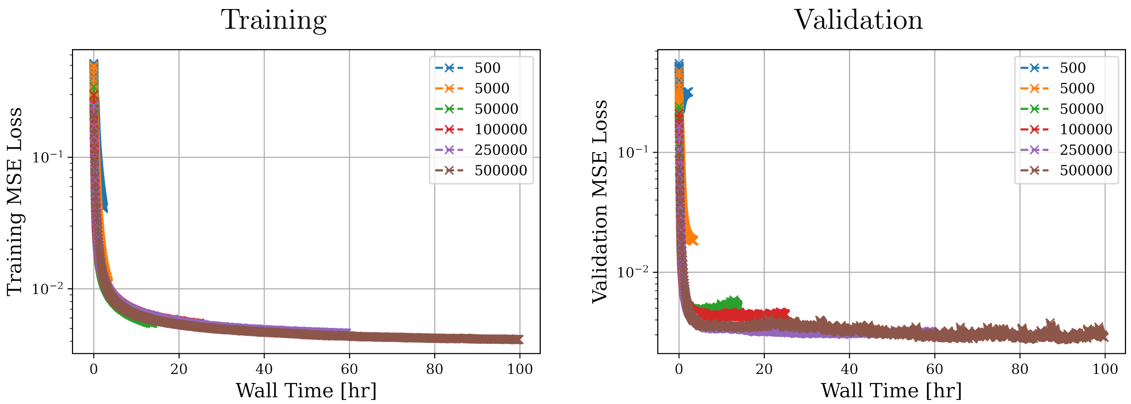


Figure 5-2: The effect of different training set sizes on the training and validation losses when training the Separation U-Net.

training set can lead to overfitting and poor generalization. Conversely, assembling a large training set can be challenging in practice, especially when dealing with complex systems and hard-to-capture data. In this scenario, since synthetic signals are used, we can generate a sufficiently large dataset to better characterize the different scales of the dataset for this problem. As shown in Fig. 5-2, a training set of fewer than $10,000$ realizations (each being 2560-long) leads to poor validation loss. This characterization with different training set sizes serves as a valuable indicator, providing us with an estimated order of magnitude for the dataset size we should strive to collect. In particular, it reinforces the importance of having sufficiently large datasets for training our deep learning models.

**Batch size** is another key factor in training dynamics. Larger batch sizes can cycle through the entire training set in fewer steps, capitalizing on the acceleration provided by parallel computation on GPUs. Conversely, smaller bath sizes are believed to exert a regularization effect, potentially leading to better model generalization [75]. Early deep learning research proposes that optimal batch sizes typically range from 1 to a few hundred, with 32 often cited as a good default value [76]. However, it is important to note that the maximum feasible batch size is constrained by the available GPU memory.

Fig. 5-3 shows the training and validation losses as batch size varies. Notably, in this scenario, variations in batch sizes do not yield significantly large changes in performance, although larger batch sizes tend to lead to faster training times and faster convergence.

The **kernel size** for the first convolutional layer is a key consideration in our problem, as detailed in the earlier chapters, serving as an important architectural modification.

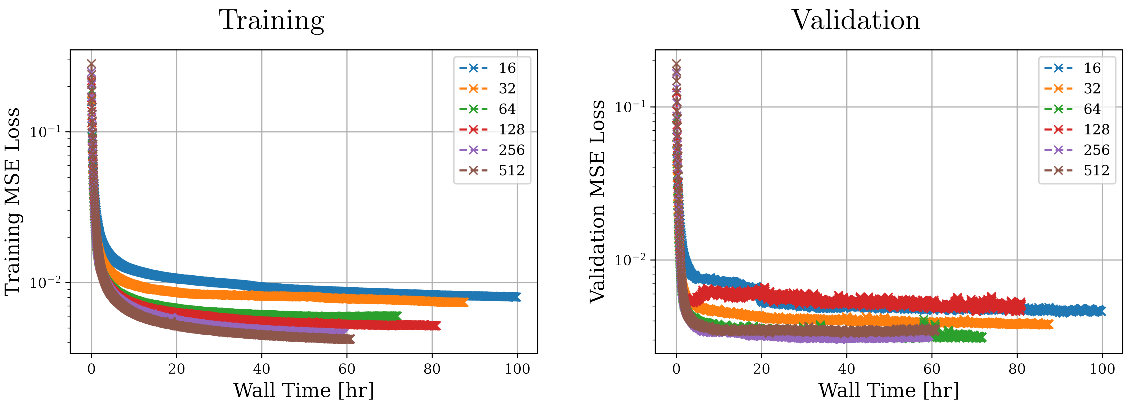


Figure 5-3: The effect of different mini-batch sizes on the training and validation losses when training the Separation U-Net.

This parameterization directly impacts the structural priors we introduce into the model, which in turn influences the representation power and training dynamics and, ultimately, the performance of our trained models.

We conduct an ablation study on this key parameterization choice, evaluating the models' performance against different sizes of first-layer kernels. Fig. 5-4 depicts the training and validation losses for the different choices of first-layer kernel sizes.

Further evaluation of the models is conducted by looking at their performance in terms of both MSE and BER, measured against different SIR levels. Fig. 5-5 showcases these results for different first-layer kernel sizes, reflecting performance enhancements for kernel sizes greater than $K = 71$. This finding is particularly interesting when contextualized against the discussion from Subsection 4.5.2, regarding the effective correlation lengths of our signal components. In particular, recalling that the true (but unobserved) parameters OFDM component, having sufficiently large window lengths—in this case, exceeding this correlation length—can lead to improved performance. This insight reinforces the importance of appropriately selecting architectural parameters in relation to the inherent characteristics of the signals at hand.

## 5.2 End-to-End Separation versus Demodulation Approaches

So far, we have been focused on the problem as "signal separation", i.e., estimating the SOI waveform $\boldsymbol{s}$, and subsequently using standard post-processing tools. We acknowledge that such a two-step approach—separation then demodulation (by treating any residual

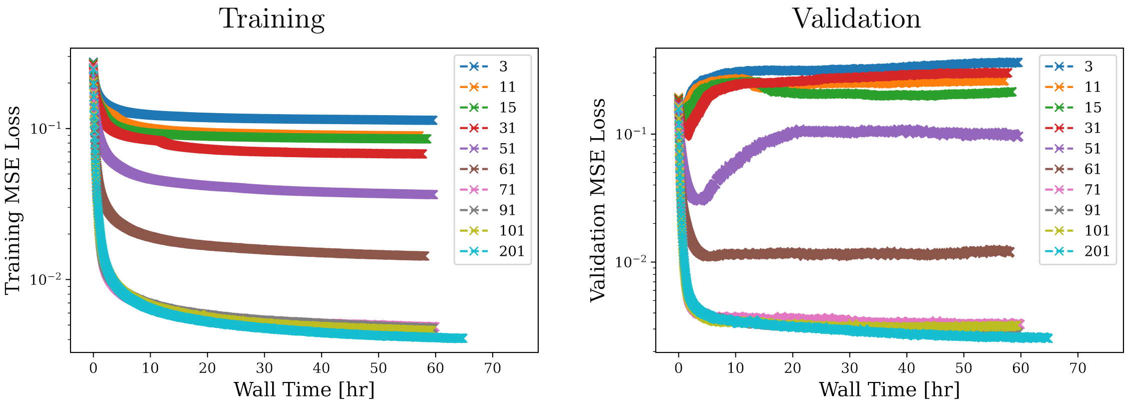


Figure 5-4: The effect of different first-layer kernel sizes on the training and validation losses when training the Separation U-Net.

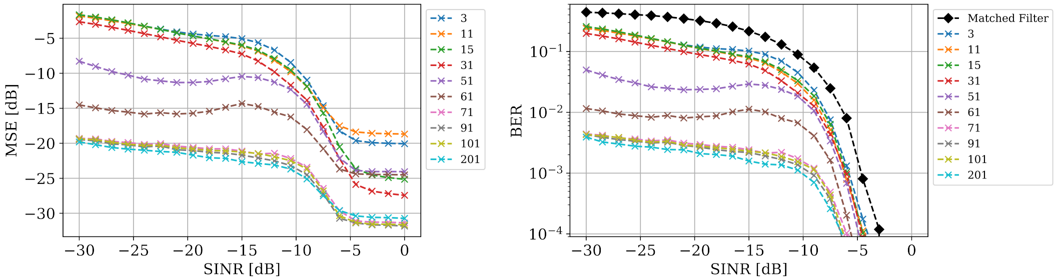


Figure 5-5: The effect of first-layer kernel size on the test performance, in terms of MSE and BER, across different SIR levels.

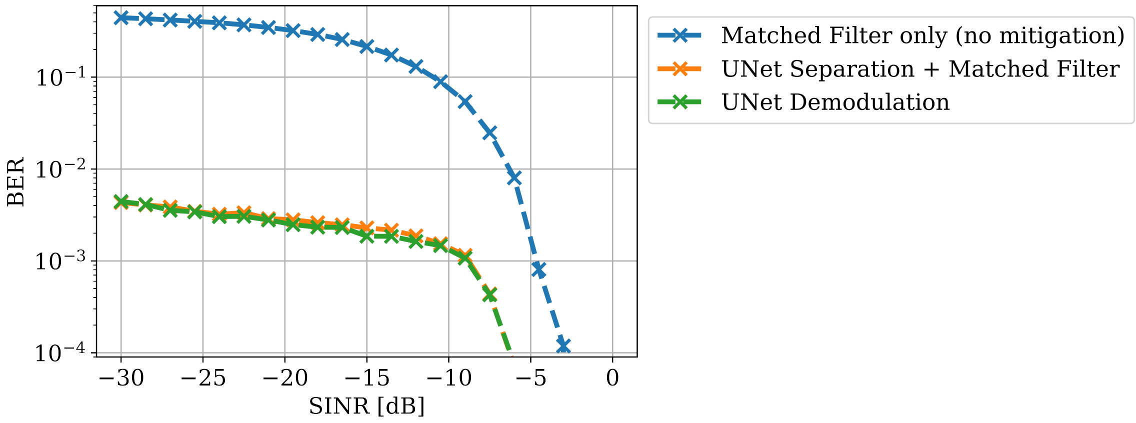


Figure 5-6: Comparison of Neural Network Methods—End-to-End Separation (using Regression to Time-Domain waveform before using Matched Filtering) versus End-to-End Demodulation (to output bitwise log probability, and subsequently using hard thresholding to recover the estimated bitstring).

interference as noise)—may not necessarily be jointly optimal. It would be of interest to investigate how an end-to-end demodulator might perform instead.

One such approach to achieve this involves modifying the U-Net by adding an extra layer that outputs a vector of logits, corresponding to the number of bits expected from $\boldsymbol{y}$. Such a network can be trained end-to-end with paired data of mixtures and corresponding bit strings from $\boldsymbol{s}$, with a training loss function corresponding to the binary cross-entropy for each bit output. Fig. 5-6 compares the BER of the two-step approach with the end-to-end demodulator approach (where hard thresholding on the logits is used to estimate the bit string). The performance difference between these two methods is found to be small, which could be attributed to the use of similar neural architectures in both cases. It is conceivable that an end-to-end demodulator may achieve different performance results if an alternative neural architecture were to be adopted for this task.

Nevertheless, it is worth noting that such an end-to-end trained model can be restrictive in its utility. For instance, if we intend to change the task at hand, such as shifting from demodulation to performing modulation classification or anomaly detection through RF fingerprinting, the applicability of a trained end-to-end demodulator would be limited. On the other hand, a trained end-to-end source separator retains relevance, as it can still provide estimated source components of presumably higher fidelity for further downstream processing. This highlights the rationale behind our emphasis on source separation as a more versatile tool applicable across various RF-related tasks.

## 5.3 A Need for Benchmarks

Thus far in this chapter, we identified configurations for the deep learning pipeline (i.e., neural architecture parameters and training hyperparameters) from extensive testing on one specific case study, involving RRC QPSK SOI in the presence of an OFDM interference. Moving forward, it is critical to assess the practical applicability of the abovementioned methods in a broader context, and in particular, beyond synthetic signals and simple signal models.

Over-the-air RF signals ("real world" signals) are generally subject to many intricate factors that may not be faithfully replicated or modeled in simulations. Testing the proposed methods on such signals will provide constructive perspectives into the relevance and improvements that deep learning methods can offer.

Furthermore, as the field of deep learning-based signal processing and source separation continues to evolve, novel algorithms and strategies will be proposed. It would hence be imperative to compare and understand which scenarios favor one method over another. Currently, there is a lack of a meaningful benchmark for single-channel signal separation with RF signals. To address this gap, the next chapter discusses the establishment of a dataset and accompanying challenge statements that can serve as a benchmark for single-channel signal separation of RF signals.

# Chapter 6

# Computational Experiments with Real-World RF Signals

In this chapter, we evaluate the efficacy of our proposed deep learning approach using real-world recordings of RF signals. The objective is to determine whether such learning-based methods can effectively improve single-channel source separation performance in practical scenarios.

To accomplish this, we test our methods on real-world signal recordings. For this effort, various types of RF waveforms have been collated and curated to form a dataset, which is then used to create the "RFChallenge" [22].[1]

We discuss the community tools and resources that we have consolidated under the umbrella of the RFChallenge. This includes introducing challenge statements, benchmarks, and performance curves, which would aid in the comparative evaluation of deep learning techniques within this field. Additionally, we present results obtained from applying our U-Net neural separation techniques to these RF recordings. These initial results offer tangible evidence for the potential of our proposed techniques in real-world settings. Importantly, the proposed U-Net approach establishes a benchmark performance for deep learning-based approaches, thereby providing a key reference point for future works in this domain.

Finally, we also address the technical challenges associated with the RFChallenge, illuminating potential areas for future exploration and research. We hope these tools, findings, and insights will further the development and application of deep learning techniques in

[1] The raw dataset was provided to us by our collaborators at Group 62, MIT Lincoln Laboratory, as part of the DAF-MIT AI Accelerator research program.

signal separation, and, more broadly, applications in RF systems.

## 6.1 RFChallenge with RF Signal Recordings

### Dataset

In an effort to facilitate the standardized evaluation of single-channel source separation methods with consideration to complex real-world settings, we dedicate efforts towards preparing and curating datasets of recorded waveforms originating from various sources. These recordings have since been used to create the "RFChallenge" [22]. The current iteration of the challenge features signals from four distinct sources—a man-made electromagnetic interference produced by equipment commonly found in modern households (EMISignal1), two types of digital communication waveforms utilized by unmanned aerial vehicles (CommSignal2 and CommSignal3) recorded over-the air, and a 5G-compliant digital communication waveform (CommSignal5G1) collected in a wired laboratory setup.

Fig. 6-1 shows a time domain and a spectrogram representation of examples across the four classes of waveforms in the dataset. The respective frames were extracted from the RF recordings and scaled to unit power on average within each dataset. Additionally, signals in EMISignal1 and CommSignal5G1 have been shifted in frequency such that the majority of their energy lies in baseband frequencies.

Further details regarding the origins of these sources are not provided in the RFChallenge as the generative processes of these signals are not explicitly known at the time of data collection. Employing human domain expertise to investigate the identities of these signals, while possible, could prove impractical as the scale of the data collection expands. Such an approach would be prohibitively resource-intensive, as attention to individual datasets is required. Therefore, it is essential that the pipeline relies as minimally as possible on user-provided specifications of the signals. Instead, the specifications of different signal types should be effectively captured by the corresponding data-driven model used in the signal separation or interference mitigation step, rather than being explicitly defined by the user.

### Problem Statements

The key challenge we identified is the processing of co-channel signals, for which components are overlapping, either partially or fully, in time and frequency. Particularly, we are

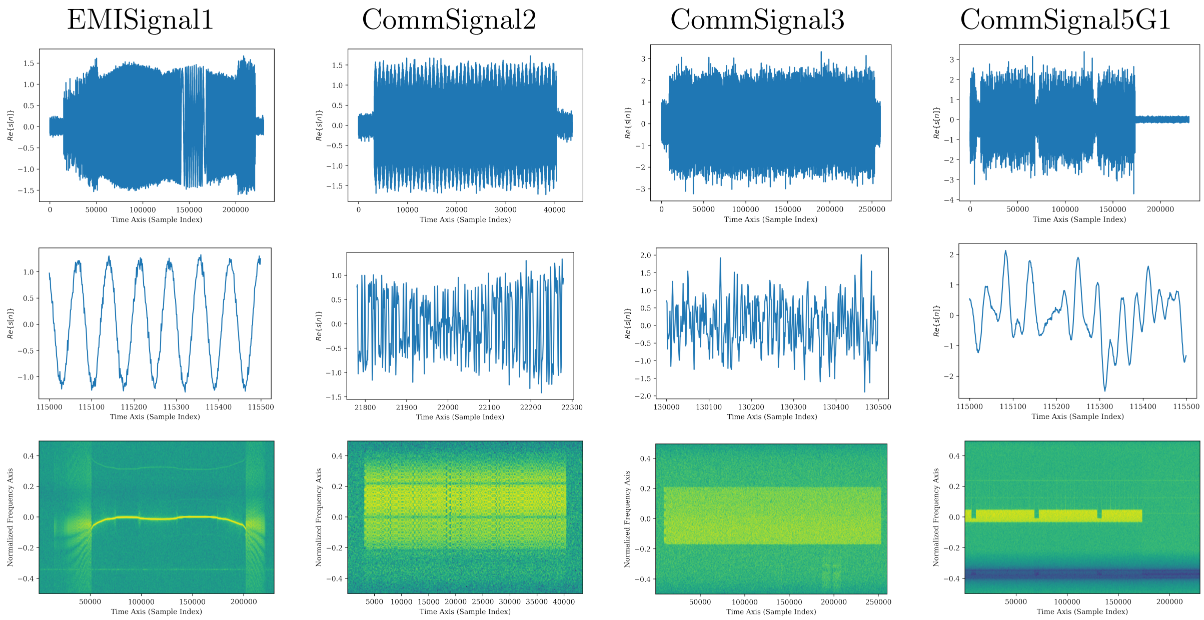


Figure 6-1: Representative frames of the dataset provided in the RFChallenge (Single-Channel Source Separation)—plotting the real part of the signal, a zoomed-in segment of the signal, and the power spectrogram of the respective waveforms.

interested in a) the separation of co-channel signals; and b) demodulation of the SOI in the presence of other interference signals.

Designing such a signal separation and/or interference mitigation tool could find uses in various applications. Having a higher fidelity estimate of the underlying component may assist in downstream processing, such as anomaly detection or finer-grained classification. From the perspective of communications, such a capability could potentially serve as an add-on to channel equalization steps before standard demodulation and decoding steps, by exploiting knowledge about the likely interference present (as trained from examples).

Based on these datasets, we are able to create a series of test cases involving mixtures of different RF signals. For our evaluation, we consider two different sub-challenges—

1. "Demodulation Sub-Challenge": We consider mixture signals comprising a single-carrier SOI and an interference signal. For the SOI, we suppose that the generation model is fully specified; we focus on the case where the SOI is a digital communication signal (one of four particular configurations, Fig. 6-2) that has been corrected for time and frequency offsets. The interference under consideration is one of the four types in the RFChallenge dataset. The key performance measures are the reconstruction

quality of the SOI (in terms of MSE) and BER from extracting the information bits.

2. "Separation Sub-Challenge": We consider mixture signals comprising an SOI and an interference signal. The SOI corresponds to a realization from the CommSignal2 dataset of the RFChallenge; this means that we do not have explicit specifications about the signal we hope to extract, but have sample realizations of it. The interference under consideration is one of the remaining three types in the RFChallenge dataset. In addition to reconstruction quality (in terms of MSE), we measure the fidelity of the estimated SOI by putting it through a black-box demodulation block/cyclic redundancy check (CRC), and reporting the CRC success rate.

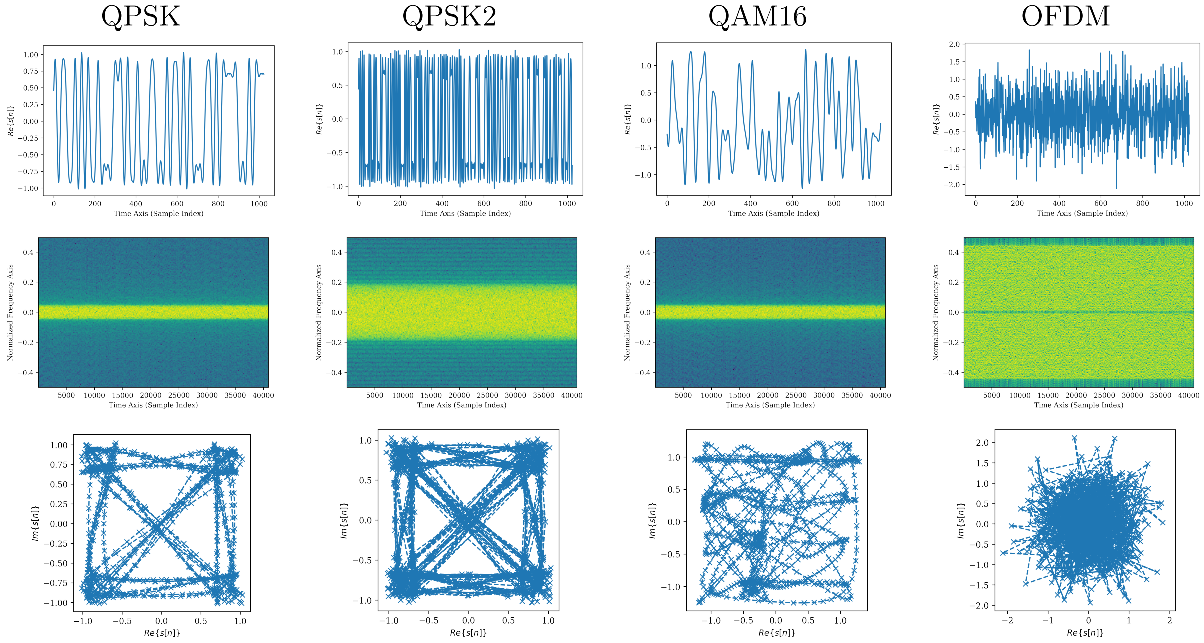


Figure 6-2: Representative visualizations of the SOI under considerations for the Demodulation Sub-Challenge—plotting the real part of the signal, the accompanying power spectrogram, and the IQ diagram for the respective SOI types.

It should be noted that in these problems, the examples we have for $\boldsymbol{b}$ (and, for the Separation Sub-challenge, $\boldsymbol{s}$) are recordings subject to impose and radio channel nonidealities. For the challenge, we are focused on offline, non-causal testing, where the data at inference is drawn from recordings of similar conditions as the training dataset.

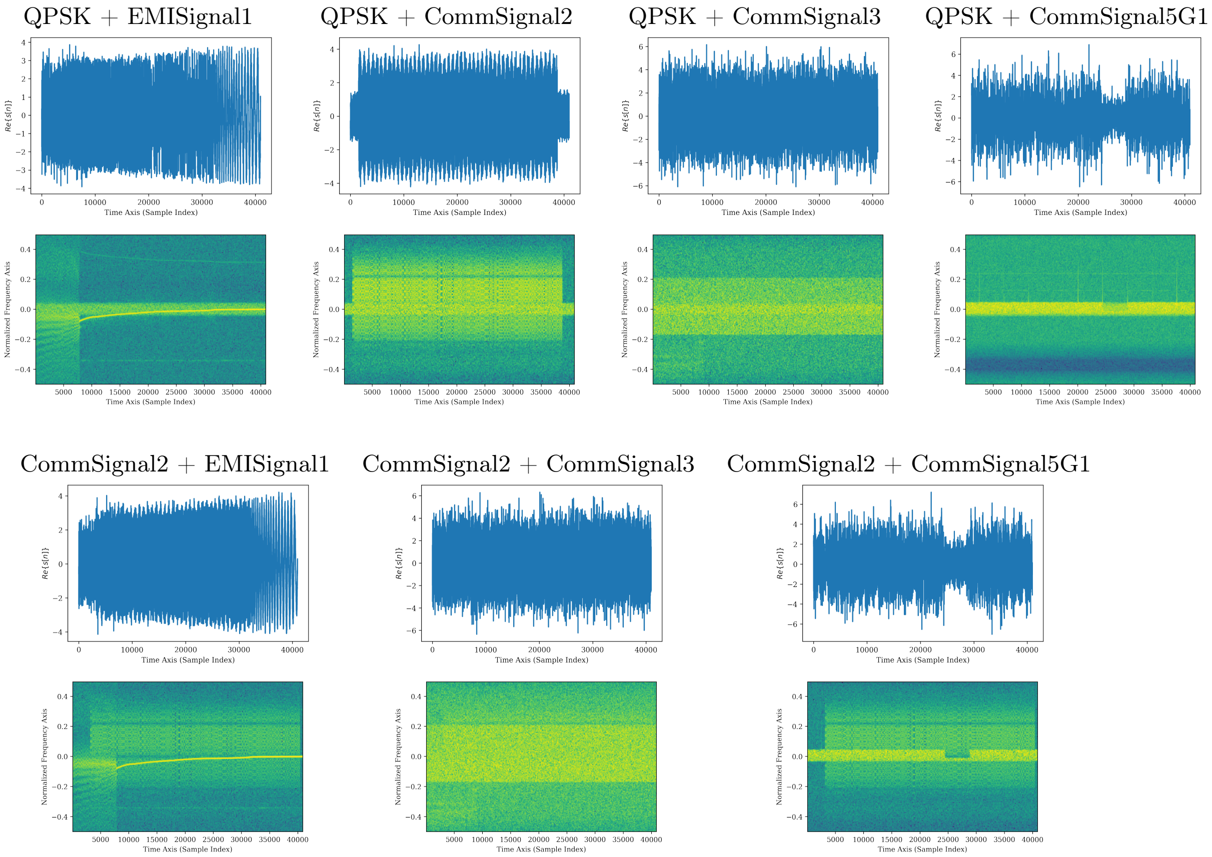


Figure 6-3: Representative visualizations of the signal mixtures in the RFChallenge Single-Channel Source Separation—plotting the real part of the signal mixture and the corresponding power spectrogram.

## 6.2 Implementation Details

We now present the details about the proposed baseline method, which is founded on the U-Net separation approach detailed in prior chapters. Keras and Tensorflow 2 are used to implement and train the U-Net [72,73]. For each RFChallenge case, we use a training set size of 240,000, for which 90% is used to train the neural network, and the remaining 10% is set aside as the validation set. The dataset contains mixtures with SIR levels drawn uniformly on the dB scale (i.e., log scale) between −33 dB and 3 dB. We train each neural network on a computing cluster with Intel Xeon Gold 6248, 192 GB RAM, and 2× NVidia Volta V100 GPU. The models are trained over a 96-hour time window, and the weights corresponding to the best validation loss are saved to prevent overfitted models. The Adam optimizer [74] with a fixed learning rate of 0.0003 and a batch size of 64 is used. [2]

[2] The GPU memory size dictates the maximum batch size that can be used for training inputs of length 40,960 samples.

## 6.3 Baseline Results for Demodulation Sub-Challenge

In this sub-challenge, we look into 4 different configurations of the SOI (Fig. 6-2). Beyond the previously explored configuration for RRC QPSK signals, we also investigate different constellations, oversampling rates, and modulation schemes (i.e., multi-carrier OFDM).

A single-carrier RRC signal with a lower sampling rate implies a broader bandwidth. This potentially results in a larger time-frequency overlap and fewer temporal redundancies or shorter correlations. By considering a QPSK signal with 4× oversampling factor (as opposed to the former's 16×)—which we call "QPSK2"—we seek to recover four times as many bits/symbols from the same observation window.

We also consider a QAM16 configuration with a 16× oversampling factor. This configuration encodes more bits per symbol while preserving the same symbol rate. We note that the accuracy of recovering the bit string also depends on precise amplitude recovery, and not just solely good phase recovery.

Lastly, we consider an OFDM SOI waveform, which is representative of modern digital communication waveforms in more complex systems. Previous chapters underscored the importance of carefully chosen neural architectures to accurately learn and capture OFDM structures. This test case is particularly relevant to solutions for a broader class of RF signals.

Collectively, these 4 configurations pose a broad spectrum of challenges. Fig. 6-4 and 6-5 show how our U-Net signal separation performs in signal reconstruction and in demodulating the underlying information, thereby reflecting its usefulness in mitigating the interference. Notably, our proposed separation procedure consistently outperforms linear MMSE estimation and standard demodulation procedures.

It is worth noting that to demodulate the SOI after signal separation, we used matched filtering for the first three waveforms with RRC pulse shaping function, and a standard demodulation based on the FFT operation for the OFDM signal. It is a topic of future investigation as to whether improved interference mitigation can be achieved beyond conventional demodulation through a combined approach, as opposed to the two-step strategy demonstrated in this work, thereby extending and generalizing the results discussed in Section 5.2.

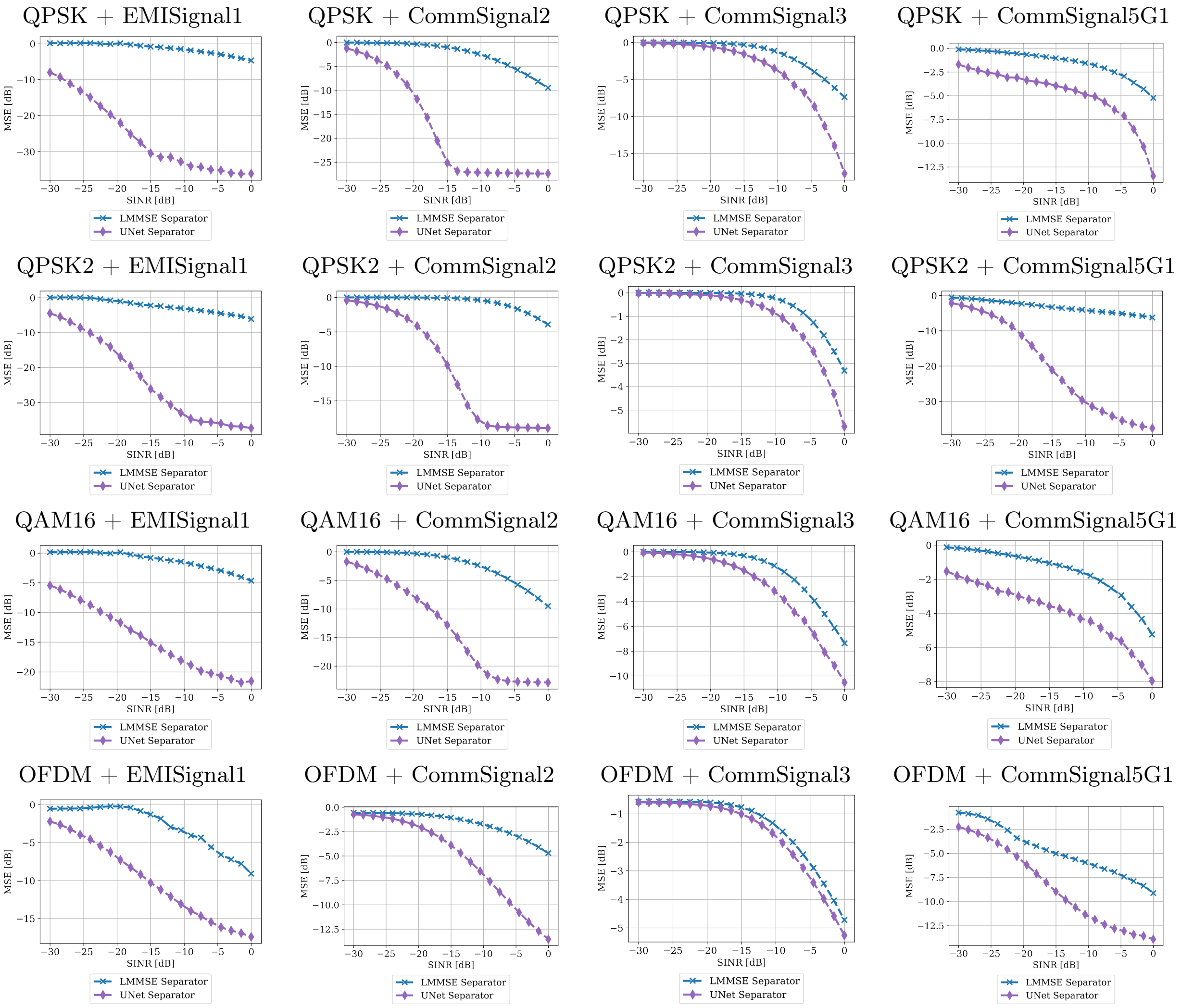


Figure 6-4: Comparison of MSE in the signal extraction and reconstruction for the single-carrier communication SOI for the Demodulation Sub-Challenge.

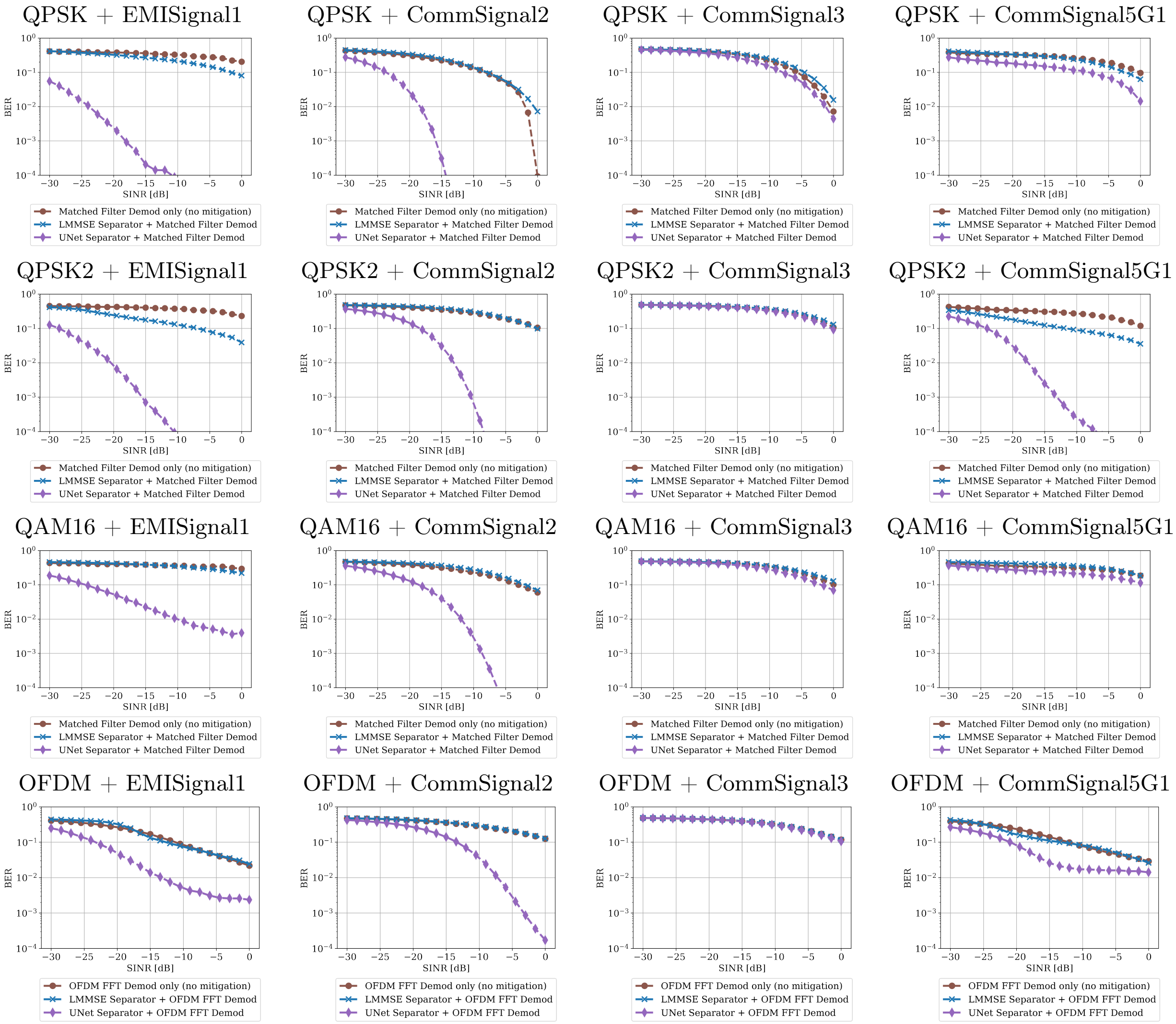


Figure 6-5: Comparison of BER in the interference mitigation for the single-carrier communication SOI for the Demodulation Sub-Challenge.

## 6.4 Baseline Results for Separation Sub-Challenge

In this sub-challenge, we consider mixtures with CommSignal2 as the SOI. Notably, we lack explicit information about its origins and specifications, a situation which differentiates this from the Demodulation Sub-Challenge described earlier. Recall that in the previous case, we had explicit SOI specifications, allowing for a more informed approach to extract and demodulate the underlying bits.

Fig. 6-6 shows how our U-Net signal separation performs in signal reconstruction, demonstrating its generally improved performance compared to the linear MMSE estimator.

Beyond the signal reconstruction, we are also interested in whether the estimated waveform can retain the integrity of the underlying structure. After inspecting the dataset with the data providers, we understand that the CommSignal2 waveforms correspond to single-frame recordings. Furthermore, the data providers have provided means of performing a cyclic redundancy check on a given signal recording to evaluate packet fidelity. [3]

Fig. 6-7 demonstrates how the signal extracted from the U-Net separation performs in this CRC procedure, thereby reflecting its relevance in interference mitigation and preservation of underlying information integrity. These observations demonstrate the versatility of the signal separation approach—by obtaining high-fidelity estimations of the SOI, we facilitate the use of other downstream processing tools that may generally perform better in high SNR settings.

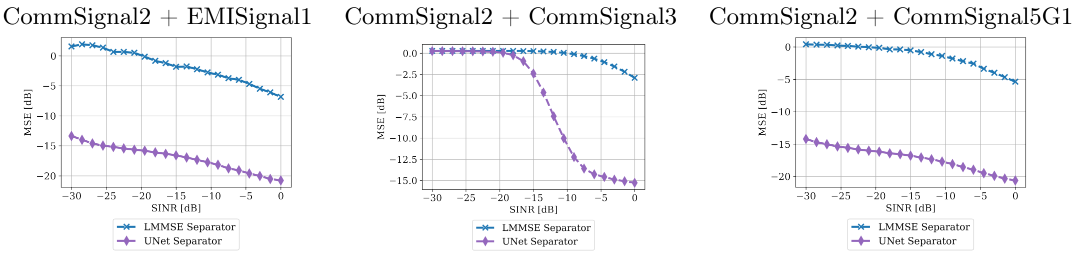


Figure 6-6: Comparison of MSE in the signal extraction and reconstruction of the CommSignal2 SOI component (whose signal model is not explicitly provided) for the Separation Sub-Challenge.

[3] In alignment with the principles set by the RFChallenge competition [22], we do not provide a thorough characterization of CommSignal2 in this thesis to uphold the integrity of the competition's data-driven premise. We direct readers to the challenge's latest publicly available resources for the most up-to-date details. It is worth noting that any insights of CommSignal2 or other data can be acquired through intensive human-in-the-loop investigation, which does not align with the scalability demands and data-driven objective of this work.

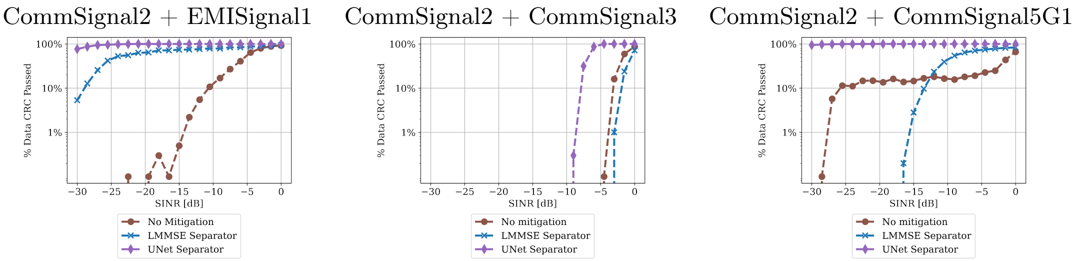


Figure 6-7: Comparison of the CRC Success Rate when a standard demodulation tool (specific to CommSignal2) is applied to the extracted signal component in the Separation Sub-Challenge.

## 6.5 Technical Challenges

The RFChallenge presents a complex task where there does not seem to be a straightforward, one-size-fits-all solution to the varied configurations discussed. There remains potential for improvement on the benchmark U-Net approach in some, if not all, of the settings. However, the task of extensive and exhaustive design for this challenge can be overwhelmingly complex.

Given the wide array of RF signals currently in use and those anticipated in future technologies, the sheer number of combinations renders the idea of a human-in-the-loop design process for each configuration impractical. Optimizing a neural network architecture or a machine learning pipeline for one specific configuration does not guarantee its generalizability across other configurations.

We also considered the effects of various hyperparameters in the previous section, but the scope of fine-tuning these for each configuration is infeasible. The complexity of the challenge is further exacerbated by the fact that we are typically interested in long time scales. For example, a 1 ms recording translates to $25,000$ samples in time for most of the settings described herein (i.e., assuming a 25 MHz sampling rate). Given the extended time scale of the RFChallenge ($40,960$ time samples), the demands on memory during training increase, as does the duration of the training process. The previous section showed extensive testing on one particular case study–QPSK + OFDM—for which a similar pipeline has been applied to various configurations with different degrees of success. However, a similar approach to an exhaustive hyperparameter search for each configuration becomes computationally taxing. Further, while different neural architectures could potentially be advantageous for different signal types (e.g., extracting a single-carrier waveform versus a multi-carrier OFDM waveform versus a waveform like CommSignal2), exploring these architectures can be costprohibitive. An ongoing effort is geared towards a more automated methodology that can efficiently handle the massive search space and yield the best achievable result.

One of the aspirations of this challenge/benchmark is to crowdsource the search for an optimal machine learning architecture—encompassing the neural architecture (or, more broadly, machine learning algorithm including non-deep learning approaches), hyperparameters, and pre-processing pipelines. The U-Net solution demonstrated here serves as a competitive benchmark. Nonetheless, opportunities remain to strive for improved performance, particularly with a smaller training set and shorter training times.

# Chapter 7

# Towards Scalability with Library of Learned Priors

Up to this point, we have focused on an end-to-end separation approach for signal mixtures. However, we note that such a strategy may encounter scalability issues as the landscape of different signal mixture types becomes more diverse. For example, the number of two-component source separation models that leverages joint statistics grows quadratically with the number of different signal types. An alternative and potentially more efficient strategy is to learn a library of independent models for each signal type, and employ the relevant trained models for the signal separation task. Such an approach could also be more amenable to address a more generalized source-separation problem, e.g., with more than two components or with a different mixture model, rather than retraining an entirely new end-to-end separation model.

Progress in deep generative methods enables such an approach. However, implementing such a strategy entails two key tasks—the training of individual generative models and the effective use of these models jointly during inference.

This chapter looks into a specific form of generative modeling that aligns with the singal separation problem and our efforts thus far. Specifically, the discussion is built upon a key result from [77] that relates MMSE estimators to probability distributions. By leveraging function approximations of the MMSE estimators, we investigate how these can be used in a scalable, alternative manner to learn individual signal models, and subsequently how they can be jointly used at inference for signal separation and interference mitigation. To train individual models, we can leverage our established knowledge from training end-to-end separators as approximations of the MMSE estimators. Thereafter, we discuss how these MMSE estimators for Gaussian denoising can be used in obtaining a *maximum a posteriori* (MAP) solution for the single-channel source separation problem.

## 7.1 Information-Theoretic MMSE

We begin by presenting the main result from [77], which relates probability distribution to MMSE estimators. This relation, seen as a generalization of the classic results in connecting mutual information to MMSE (also referred to as I-MMSE relations), forms the basis for our discussion in this chapter.

Consider the following Gaussian channel

$$\boldsymbol{x}_\gamma = \sqrt{\gamma}\boldsymbol{x} + \boldsymbol{z}, \tag{7.1}$$

where $\boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I})$, and $\gamma \in \mathbb{R}^+$ corresponds to the SNR of the Gaussian channel. The main result from [77] connects the MMSE of denoising $\boldsymbol{x}_\gamma$ to the data distribution of $\boldsymbol{x}$ via

$$-\log p(\boldsymbol{x}) = \frac{d}{2}\log(2\pi e) - \frac{1}{2}\int_0^\infty \left(\frac{d}{1+\gamma} - \mathrm{mmse}(\boldsymbol{x},\gamma)\right) d\gamma \tag{7.2}$$

where $\mathrm{mmse}(\boldsymbol{x},\gamma)$ is the pointwise MMSE, defined as

$$\mathrm{mmse}(\boldsymbol{x},\gamma) \triangleq \mathbb{E}_{\mathsf{z}}\left[\|\boldsymbol{x} - \widehat{\mathbf{x}}^*\left(\boldsymbol{x}_\gamma,\gamma\right)\|_2^2\right] \tag{7.3}$$

where $\widehat{\mathbf{x}}^*\left(\boldsymbol{x}_\gamma,\gamma\right)$ is the optimal denoising function (MMSE estimator), i.e.,

$$\widehat{\mathbf{x}}^*\left(\boldsymbol{x}_\gamma,\gamma\right) \triangleq \arg\min_{\widehat{x}(\boldsymbol{x}_\gamma,\gamma)} \mathbb{E}_{\mathsf{x}_\gamma,\mathsf{z}}\left[\|\boldsymbol{x} - \widehat{x}(\boldsymbol{x}_\gamma,\gamma)\|_2^2\right]. \tag{7.4}$$

We refer to (7.4) as the MMSE denoiser (for $\boldsymbol{x}_\gamma$ at SNR level $\gamma$) to avoid any confusion with the MMSE estimator described for the signal separation problem.

From (7.2), we observe that the distribution of $\boldsymbol{x}$ can be exactly related to the MMSE denoising objective. Therefore, learning the prior for $\boldsymbol{x}$ is akin to learning the MMSE denoisers of (7.1) over all possible SNR values.

In the following section, we investigate how these priors (and their corresponding MMSE denoisers) factor into the source separation problem. On the other hand, we acknowledge that the optimal MMSE denoisers may not be analytically tractable beyond relatively simple settings (e.g., Gaussian assumption on $\boldsymbol{x}$)—which we address in our subsequent discussion.

## 7.2 MAP Estimation Approach to Single-Channel Source Separation

We now return our attention to the mixture model, given by

$$\boldsymbol{y} = \boldsymbol{s} + \kappa \boldsymbol{b}, \tag{7.5}$$

where $\boldsymbol{s}$, $\boldsymbol{b}$ are the (unobserved) statistically independent components that make up our received mixture $\boldsymbol{y}$, and $\kappa \in \mathbb{R}^+$ corresponds to the SIR level in this mixture signal. As previously established, under our problem formulation, we do not have access to the true signal models for $\boldsymbol{s}$ and $\boldsymbol{b}$. However, the previous insights suggest that we can have an approximation of these models in the form of (7.2), assuming that we can derive exact or approximate MMSE denoisers of $\boldsymbol{s}$ and $\boldsymbol{b}$. In this context, we aim to address the inference case where, given an unseen test signal $\boldsymbol{y}$, how do we utilize these learned priors (in the form of potentially good approximations of $p_{\mathsf{s}}(\boldsymbol{s})$ and $p_{\mathsf{b}}(\boldsymbol{b})$) to estimate the underlying source signals—particularly, the SOI, $\boldsymbol{s}$.

To capitalize on the availability of these individual priors, we look toward the Bayesian framework and seek the MAP estimator of $\boldsymbol{s}$,

$$\arg\min_{\boldsymbol{s}} -\log p_{\mathsf{s}|\mathsf{y}}(\boldsymbol{s}|\boldsymbol{y}) \tag{7.6}$$

$$= \arg\min_{\boldsymbol{s}} -\log p_{\mathsf{s}}(\boldsymbol{s}) - \log p_{\mathsf{b}}\left(\frac{\boldsymbol{y}-\boldsymbol{s}}{\kappa}\right) \tag{7.7}$$

$$\begin{split} = \arg\min_{\boldsymbol{s}} \left(\frac{d}{2}\log(2\pi e) - \frac{1}{2}\int_0^\infty \left(\frac{d}{1+\gamma_1} - \mathrm{mmse}(\boldsymbol{s},\gamma_1)\right) d\gamma_1\right) + \dots \\ \left(\frac{d}{2}\log(2\pi e) - \frac{1}{2}\int_0^\infty \left(\frac{d}{1+\gamma_2} - \mathrm{mmse}(\boldsymbol{b},\gamma_2)\right)\Bigg|_{\boldsymbol{b} \,=\, (\boldsymbol{y}-\boldsymbol{s})\,/\kappa} d\gamma_2\right) \end{split} \tag{7.8}$$

$$= \arg\min_{\boldsymbol{s}}\ C + \frac{1}{2}\left(\int_0^\infty \mathrm{mmse}(\boldsymbol{s},\gamma_1) d\gamma_1 + \int_0^\infty \mathrm{mmse}(\boldsymbol{b},\gamma_2)|_{\boldsymbol{b} \,=\, (\boldsymbol{y}-\boldsymbol{s})\,/\kappa}\, d\gamma_2\right), \tag{7.9}$$

and similarly, by symmetry, we can arrive at a similar expression for $\arg\min_{\boldsymbol{b}} -\log p_{\mathsf{b}|\mathsf{y}}(\boldsymbol{b}|\boldsymbol{y})$. In (7.8), we recall the pointwise MMSE, $\mathrm{mmse}(\cdot,\cdot)$, to be an expectation over Gaussian noise realization, as presented earlier in (7.3). To get from (7.8) to (7.9), we collate all additive terms that do not depend on $\boldsymbol{s}$; therefore, we are mainly interested in minimizing the quantity within the parenthesis of (7.9). Note that computing the improper integrals is not the focus of this optimization problem, but rather identifying the value of $\boldsymbol{s}$ that corresponds to the minimum point of the above objective function.

To gain further insights into this approach, we explore two particular scenarios. Firstly, we examine mixtures of multivariate Gaussians for the signal components, for which the MMSE denoiser can be analytically characterized, thereby serving as a sanity check for the efficacy of (7.9) for estimating $\boldsymbol{s}$. Secondly, we analyze the case where we use a data-driven approach for the MMSE denoisers, using them in a Monte Carlo approximation to perform the optimization (7.9). This case illuminates the opportunities and challenges associated with such an approach, guiding avenues for future extensions of this line of work.

### 7.2.1 Case Study with Multivariate Gaussian

We begin by studying an easier configuration where the analytical form of the MMSE denoiser can be derived. The purpose of this discussion is to serve as a sanity check in ensuring that the procedure above, particularly finding the stationary points of (7.9), indeed yields the MAP estimate of $\boldsymbol{s}$ from $\boldsymbol{y}$.

For this analysis, we consider $\boldsymbol{s}$ and $\boldsymbol{b}$ to be multivariate Gaussian, i.e.,

$$\boldsymbol{s} \sim \mathcal{CN}(\mu_s, \Sigma_s) \ , \ \boldsymbol{b} \sim \mathcal{CN}(\mu_b, \Sigma_b)$$

(and thus, $\boldsymbol{y}$ also follows a multivariate Gaussian distribution). Recall that the MMSE estimate of $\boldsymbol{s}$ upon observing $\boldsymbol{y}$, which is linear, is given by

$$\widehat{\boldsymbol{s}}_{\text{MMSE}}(\boldsymbol{y}) = \Sigma_s \left(\Sigma_s + \kappa^2 \Sigma_b\right)^{-1} (\boldsymbol{y} - \mu_s - \kappa \mu_b) + \mu_s. \tag{7.10}$$

(We also note that

$$\widehat{\boldsymbol{b}}_{\text{MMSE}}(\boldsymbol{y}) = \frac{\boldsymbol{y} - \widehat{\boldsymbol{s}}_{\text{MMSE}}(\boldsymbol{y})}{\kappa} = \kappa \Sigma_b \left(\Sigma_s + \kappa^2 \Sigma_b\right)^{-1} (\boldsymbol{y} - \mu_s - \kappa \mu_b) + \mu_b, \tag{7.11}$$

which reflects the symmetry in the roles between the two terms.) In the Gaussian case, the LMMSE estimator (which corresponds to the conditional mean) coincides with the MAP estimator (corresponding to the mode of the conditional distribution $p_{\mathsf{s}|\mathsf{y}}(\boldsymbol{s}|\boldsymbol{y})$). As part of this validation, we seek to verify if the stationary point(s) of (7.9) (using the MMSE denoisers for $\boldsymbol{s}$ and $\boldsymbol{b}$ respectively) yield the same solution as in (7.10).

We now proceed with the analysis for mixtures of multivariate Gaussian components. First, we establish the optimal MMSE estimator of a Gaussian random variable $\boldsymbol{x} \sim \mathcal{CN}(\mu_x, \Sigma_x)$ from the observation model (7.1), which can be expressed analytically as

$$\widehat{\mathbf{x}}^* \left(\boldsymbol{x}_\gamma, \gamma\right) = \sqrt{\gamma}\Sigma_x \left(\gamma\Sigma_x + \boldsymbol{I}\right)^{-1} \left(\boldsymbol{x}_\gamma - \sqrt{\gamma}\mu_x\right) + \mu_x. \tag{7.12}$$

As such, the pointwise MMSE (7.3) in this scenario can be expressed as

$$\mathrm{mmse}(\boldsymbol{x}, \gamma) = \mathbb{E}_{\boldsymbol{z}} \left[ \|\boldsymbol{x} - \sqrt{\gamma}\Sigma_x \left(\gamma\Sigma_x + \boldsymbol{I}\right)^{-1} (\underbrace{\sqrt{\gamma}\boldsymbol{x} + \boldsymbol{z}}_{\boldsymbol{x}_\gamma} - \sqrt{\gamma}\mu_x) - \mu_x\|_2^2 \right]. \tag{7.13}$$

We can rewrite (7.13) for both signal components under study (the SOI $\boldsymbol{s}$ and interference $\boldsymbol{b}$), corresponding to

$$\mathrm{mmse}(\boldsymbol{s}, \gamma_1) = \mathbb{E}_{\boldsymbol{z}_1 \sim \mathcal{N}(0, \boldsymbol{I})} \left[ \|\boldsymbol{s} - \sqrt{\gamma_1}\Sigma_s \left(\gamma_1\Sigma_s + \boldsymbol{I}\right)^{-1} \left(\sqrt{\gamma_1}\boldsymbol{s} + \boldsymbol{z}_1 - \sqrt{\gamma_1}\mu_s\right) + \mu_s\|_2^2 \right], \tag{7.14}$$

$$\mathrm{mmse}(\boldsymbol{b}, \gamma_2) = \mathbb{E}_{\boldsymbol{z}_2 \sim \mathcal{N}(0, \boldsymbol{I})} \left[ \|\boldsymbol{b} - \sqrt{\gamma_2}\Sigma_b \left(\gamma_2\Sigma_b + \boldsymbol{I}\right)^{-1} \left(\sqrt{\gamma_2}\boldsymbol{b} + \boldsymbol{z}_2 - \sqrt{\gamma_2}\mu_b\right) + \mu_b\|_2^2 \right]. \tag{7.15}$$

Further, we recall that our objective function is given by (7.9). We omit the terms that do not depend on $\boldsymbol{s}$, expressing the objective function simply as

$$\mathcal{L}(\boldsymbol{s}) = \frac{1}{2}\int_0^\infty \mathrm{mmse}(\boldsymbol{s}, \gamma_1) d\gamma_1 + \frac{1}{2}\int_0^\infty \mathrm{mmse}(\boldsymbol{b}, \gamma_2)\Big|_{\boldsymbol{b} = \left(\frac{\boldsymbol{y} - \boldsymbol{s}}{\kappa}\right)} d\gamma_2. \tag{7.16}$$

For notation convenience, let the corresponding MMSE denoising linear filters be denoted as

$$W_s \triangleq \sqrt{\gamma_1}\Sigma_s \left(\gamma_1\Sigma_s + \boldsymbol{I}\right)^{-1}$$
$$W_b \triangleq \sqrt{\gamma_2}\Sigma_b \left(\gamma_2\Sigma_b + \boldsymbol{I}\right)^{-1}.$$

Next, we are interested in finding the stationary point of (7.16). To do so, we compute the gradient of the objective function with respect to $\boldsymbol{s}$, thereby obtaining

$$\begin{aligned}\nabla_{\boldsymbol{s}}\mathcal{L}(\boldsymbol{s}) = \nabla_{\boldsymbol{s}}\, \frac{1}{2}\Bigg(&\int_0^\infty \mathbb{E}_{\boldsymbol{z}_1}\left[\|\boldsymbol{s} - W_s(\sqrt{\gamma_1}\boldsymbol{s} + \boldsymbol{z}_1 - \sqrt{\gamma_1}\mu_s) + \mu_s)\|_2^2\right] d\gamma_1 + \dots \\ &\int_0^\infty \mathbb{E}_{\boldsymbol{z}_2}\left[\|\left(\frac{\boldsymbol{y}-\boldsymbol{s}}{\kappa}\right) - W_b(\sqrt{\gamma_2}\left(\frac{\boldsymbol{y}-\boldsymbol{s}}{\kappa}\right) + \boldsymbol{z}_2 - \sqrt{\gamma_2}\mu_b) + \mu_b)\|_2^2\right] d\gamma_2\Bigg)\,. \end{aligned} \tag{7.17}$$

Through some algebraic manipulation and resolving the gradient with respect to $\boldsymbol{s}$, we obtain the following expression

$$\begin{aligned}\nabla_{\boldsymbol{s}}\mathcal{L}(\boldsymbol{s}) = \frac{1}{2}\Bigg(&\int_0^\infty \mathbb{E}_{\boldsymbol{z}_1}\left[\nabla_{\boldsymbol{s}}\|(\boldsymbol{I} - \sqrt{\gamma_1}W_s)\boldsymbol{s}(\boldsymbol{I} - \sqrt{\gamma_1}W_s)\mu_s - W_s\boldsymbol{z}_1)\|_2^2\right] d\gamma_1 + \dots \end{aligned}\tag{7.18}$$

$$\begin{aligned}&\int_0^\infty \mathbb{E}_{\boldsymbol{z}_2}\left[\nabla_{\boldsymbol{s}}\|(\boldsymbol{I} - \sqrt{\gamma_2}W_b)\left(\frac{\boldsymbol{y}-\boldsymbol{s}}{\kappa}\right) - (\boldsymbol{I} - \sqrt{\gamma_2}W_b)(\mu_b) - W_b\boldsymbol{z}_2\|_2^2\right] d\gamma_2\Bigg) \\ &= \int_0^\infty \mathbb{E}_{\boldsymbol{z}_1}\left[(\boldsymbol{I} - \sqrt{\gamma_1}W_s)\left((\boldsymbol{I} - \sqrt{\gamma_1}W_s)\boldsymbol{s} - (\boldsymbol{I} - \sqrt{\gamma_1}W_s)\mu_s - W_s\boldsymbol{z}_1)\right)\right] d\gamma_1 - \dots\end{aligned}\tag{7.19}$$

$$\begin{aligned}&\frac{1}{\kappa}\int_0^\infty \mathbb{E}_{\boldsymbol{z}_2}\left[(\boldsymbol{I} - \sqrt{\gamma_2}W_b)\left((\boldsymbol{I} - \sqrt{\gamma_2}W_b)\left(\frac{\boldsymbol{y}-\boldsymbol{s}}{\kappa}\right) - (\boldsymbol{I} - \sqrt{\gamma_2}W_b)(\mu_b) - W_b\boldsymbol{z}_2\right)\right] d\gamma_2 \\ &= \int_0^\infty (\boldsymbol{I} - \sqrt{\gamma_1}W_s)\left((\boldsymbol{I} - \sqrt{\gamma_1}W_s)\boldsymbol{s} - (\boldsymbol{I} - \sqrt{\gamma_1}W_s)\mu_s\right) d\gamma_1 - \dots\end{aligned}\tag{7.20}$$

$$\begin{aligned}&\frac{1}{\kappa}\int_0^\infty (\boldsymbol{I} - \sqrt{\gamma_2}W_b)\left((\boldsymbol{I} - \sqrt{\gamma_2}W_b)\left(\frac{\boldsymbol{y}-\boldsymbol{s}}{\kappa}\right) - (\boldsymbol{I} - \sqrt{\gamma_2}W_b)\mu_b\right) d\gamma_2 \\ &= \int_0^\infty (\boldsymbol{I} - \sqrt{\gamma_1}W_s)^2\,(\boldsymbol{s} - \mu_s)\, d\gamma_1 - \frac{1}{\kappa}\int_0^\infty (\boldsymbol{I} - \sqrt{\gamma_2}W_b)^2\left(\left(\frac{\boldsymbol{y}-\boldsymbol{s}}{\kappa}\right) - \mu_b\right) d\gamma_2\end{aligned}\tag{7.21}$$

We are interested in identifying the stationary points, where the gradient $\nabla_{\boldsymbol{s}}\mathcal{L}(\boldsymbol{s}) = 0$. Essentially, this entails finding value(s) of $\boldsymbol{s}$ where the two terms in (7.21) cancel out, resulting in zero gradient. Although (7.21) may initially seem complex, this expression can be further simplified by invoking particular properties of terms appearing in the integral.

To achieve this, we focus on the first integrand that is dependent on $\gamma_1$, given by

$$\boldsymbol{I} - \sqrt{\gamma_1}W_s = \boldsymbol{I} - \gamma_1\Sigma_s\left(\gamma_1\Sigma_s + \boldsymbol{I}\right)^{-1}, \tag{7.22}$$

$$\boldsymbol{I} - \sqrt{\gamma_2}W_b = \boldsymbol{I} - \gamma_2\Sigma_b\left(\gamma_2\Sigma_b + \boldsymbol{I}\right)^{-1}. \tag{7.23}$$

We also recall that $\Sigma_s$, $\Sigma_b$, as symmetric matrices, are diagonalizable, meaning that we can express them as

$$\Sigma_s = U_s\Lambda_s U_s^H\ ,\ \Sigma_b = U_b\Lambda_b U_b^H\,.$$

In addition, for a diagonalizable matrix $A = PDP^{-1}$, we have the following identifies—

$$A^k = \underbrace{(PDP^{-1})...(PDP^{-1})}_{k \text{ times}} = PD^kP^{-1},$$

and

$$A + \boldsymbol{I} = PDP^H + P\boldsymbol{I}P^H = P(D + \boldsymbol{I})P^H.$$

By applying these properties, we can simplify the LMMSE denoising filter terms. We first demonstrate it for the signal $\boldsymbol{s}$, where

$$\begin{aligned} W_s &= \sqrt{\gamma_1}\Sigma_s(\gamma_1\Sigma_s + \boldsymbol{I})^{-1} && (7.24) \\ &= U_s(\sqrt{\gamma_1}\Lambda_s)U_s^H(U_s(\gamma_1\Lambda_s)U_s^H + U_s\boldsymbol{I}U_s^H)^{-1} && (7.25) \\ &= U_s\left(\sqrt{\gamma_1}\Lambda_s(\gamma_1\Lambda_s + \boldsymbol{I})^{-1}\right)U_s^H && (7.26) \\ & && (7.27) \end{aligned}$$

and therefore, simplifying the term that appears in the integrand of (7.21)

$$\begin{aligned} (\boldsymbol{I} - \sqrt{\gamma_1}W_s)^2 &= (\boldsymbol{I} - \sqrt{\gamma_1}U_s\left(\sqrt{\gamma_1}\Lambda_s(\gamma_1\Lambda_s + \boldsymbol{I})^{-1}\right)U_s^H)^2 && (7.28) \\ &= U_s\left(\boldsymbol{I} - \gamma_1\Lambda_s(\gamma_1\Lambda_s + \boldsymbol{I})^{-1}\right)^2 U_s^H && (7.29) \\ &= U_s\left(\gamma_1\Lambda_s + \boldsymbol{I}\right)^{-2}U_s^H. && (7.30) \end{aligned}$$

Finally, with such simplification, we see that the improper integral can be analytically computed

$$\begin{aligned} \int_0^\infty (\boldsymbol{I} - \sqrt{\gamma_1}W_s)^2\,(\boldsymbol{s} - \mu_s)\,d\gamma_1 &= \int_0^\infty \left(U_s\left(\gamma_1\Lambda_s + \boldsymbol{I}\right)^{-2}U_s^H\right)(\boldsymbol{s} - \mu_s)\,d\gamma_1 && (7.31) \\ &= U_s\left(\int_0^\infty \left(\gamma_1\Lambda_s + \boldsymbol{I}\right)^{-2}d\gamma_1\right)U_s^H\,(\boldsymbol{s} - \mu_s) && (7.32) \\ &= U_s\Lambda_s^{-1}U_s^H\,(\boldsymbol{s} - \mu_s) && (7.33) \\ &= \Sigma_s^{-1}\,(\boldsymbol{s} - \mu_s) && (7.34) \end{aligned}$$

And similar expressions hold for $\boldsymbol{b}$.

We are thus able to simplify (7.21) as follows

$$\nabla_{\boldsymbol{s}}\mathcal{L}(\boldsymbol{s}) = \Sigma_s^{-1}(\boldsymbol{s}-\mu_s) - \frac{1}{\kappa}\Sigma_b^{-1}\left(\left(\frac{\boldsymbol{y}-\boldsymbol{s}}{\kappa}\right)-\mu_b\right), \tag{7.35}$$

Finaly, we seek values of $\boldsymbol{s}$ for which this gradient is zero. We observe that by substituting $\boldsymbol{s}$ with (7.10), the result is 0, indicating that the LMMSE estimate is indeed the stationary point of the optimization problem.

This result provides a sanity check on our formulation. We see that the optimization problem, framed in terms of individual denoising MMSE estimators, can lead to the MAP (or, equivalently, in the Gaussian case, the LMMSE estimate) when considered jointly.

Nonetheless, in many practical settings, we may not have access to the analytical form of the MMSE denoiser. Hence, we explore the possibility of approximating these estimators using deep neural networks.

### 7.2.2 Case Study with Learned Models

While the earlier derivation is instructive, the MMSE denoiser might not be analytically tractable in more complex scenarios (e.g., deviating from the Gaussian assumption). However, as we have explored in previous chapters, we can use deep neural networks to learn function approximators of MMSE estimators (which, in this context, are the MMSE denoisers). Indeed, this setup parallels our original problem of learning an MMSE estimator for separation, with the key insight that the interference component is now an AWGN. Recalling the relation in (7.2), access to (an approximate of) the MMSE estimator implies also having access to (an approximate of) the corresponding data distribution, which can then be applied to the signal separation problem.

In this segment, we revisit the example with an RRC QPSK signal as the SOI and an OFDM with QPSK subcarriers as the interference. We describe the process in two parts—one, training individual models, in the form of MMSE estimators in the presence of varying Gaussian noise levels (i.e., denoisers), and two, using these individual models for the signal separation problem on an unseen signal mixture to recover the unobserved signal components.

### Learning Denoising MMSE Estimators

In the first part, we approximate the individual MMSE estimators (7.4) with deep neural networks. This is achieved by training a multivariate regression model using paired examples of noisy signals and their corresponding noiseless ground-truth signals, with MSE as the loss function.

To train such a model, we adopt a U-Net architecture similar to the one described in the previous chapters. Nonetheless, in this scenario, we make a few modifications, namely—increasing the kernel sizes of all convolutional layers to 15 (while retaining the first-layer kernel size at 101), and augmenting the downsampling blocks with reshaped copies of the signal input.

We also rescale the Gaussian channel model (7.1) so that the signal component has scaled unit power on average, i.e.,

$$\boldsymbol{s}_\gamma = \boldsymbol{s} + \frac{1}{\sqrt{\gamma}}\boldsymbol{z}, \tag{7.36}$$

and similarly for the interference $\boldsymbol{b}$.

We train the MMSE denoiser using a dataset of signal examples $\boldsymbol{s}$ (and similarly for $\boldsymbol{b}$). We introduce AWGN as in (7.36) between the range of SNR from $-36$ dB and 36 dB, with values drawn uniformly on the dB (logarithmic) scale. We seek to minimize the squared error between the denoised output and the ground-truth noise-free signal. We used a training set size of $100,000$, with a batch size of 256. Note that during training, a new realization of Gaussian noise is generated at every training step, effectively augmenting the training set. We find this beneficial in avoiding overfitting, despite the significant increase in the number of parameters in the modified U-Net architecture.

We reiterate that the models for $\boldsymbol{s}$ and $\boldsymbol{b}$ are trained independently. The joint statistics of the two signals are not seen at this stage of training.

### Inference on Mixtures

Equipped with learned individual models for the signal components $\boldsymbol{s}$ and $\boldsymbol{b}$, we turn our attention to the second part of the problem, which is to use the models jointly for signal separation. The key lies in using the learned MMSE denoisers to find the minimizer of (7.9).

Algorithm 1 is based on a Monte Carlo approximation of (7.9), and it finds the minimizer using a stochastic gradient descent approach. It is worth noting that the expectation over

$\boldsymbol{z}_1, \boldsymbol{z}_2$ and the improper integrals over $\gamma_1, \gamma_2$ cannot be resolved analytically in this context; thus, we resort to a random sample drawn from their respective distributions per iteration and then compute the empirical average and sum, respectively.

Additionally, we introduce a few implementation tricks. Primarily, since we focus mainly on the lower SIR regimes, we choose to estimate $\boldsymbol{b}$ for our optimization problem, instead of $\boldsymbol{s}$, which corresponds to the larger magnitude component in our observation. Theoretically, due to symmetry, the choice of estimating $\boldsymbol{b}$ in place of $\boldsymbol{s}$ should arrive at the same solution. Another benefit of this is that we can obtain a good initialization on $\boldsymbol{b}$, where we can use its denoised version (estimate of $\boldsymbol{b}$ from $\boldsymbol{y}/\kappa$ by treating $\boldsymbol{s}$ as Gaussian noise) as a good starting point.

**Algorithm 1** Proposed Optimization based on the I-MMSE/Diffusion

```
function SEPARATION($\boldsymbol{y}$, $\kappa$, $M$, $B$, $\{\eta_i\}_{i=0}^{M-1}$, $\widehat{\boldsymbol{b}}^{(0)}$)    ▷ $M$ total steps, batch size $B$ per step, learning rate $\eta_i$ at step $i$
    $\widehat{\boldsymbol{b}}^{(0)} = \widehat{\boldsymbol{b}}_{\text{MMSE}}(\boldsymbol{y}/\kappa, \kappa^2)$    ▷ Initialization on estimated interference
    for $i \leftarrow 0, M-1$ do
        for $j \leftarrow 0, B-1$ do
            $\widehat{\boldsymbol{s}}^{(i)} = \boldsymbol{y} - \kappa\,\widehat{\boldsymbol{b}}^{(i)}$    ▷ Compute the estimated SOI
            $\gamma_1, \gamma_2 \sim \mathcal{U}\{0, 10^{3.8}\}$    ▷ Draw random SNR values from uniform distribution
            $\boldsymbol{z}_1, \boldsymbol{z}_2 \sim \mathcal{N}(0, \mathbf{I})$    ▷ Draw random Gaussian noise realizations
            $\tilde{\boldsymbol{s}} = \widehat{\boldsymbol{s}}^{(i)} + \frac{1}{\sqrt{\gamma_1}}\boldsymbol{z}_1$
            $\mathcal{L}_{s,j} = \|\widehat{\boldsymbol{s}}^{(i)} - \widehat{\boldsymbol{s}}_{\text{MMSE}}(\tilde{\boldsymbol{s}}, \gamma_1)\|_2^2$    ▷ Compute error for estimated SOI
            $\tilde{\boldsymbol{b}} = \widehat{\boldsymbol{b}}^{(i)} + \frac{1}{\sqrt{\gamma_2}}\boldsymbol{z}_2$
            $\mathcal{L}_{b,j} = \|\widehat{\boldsymbol{b}}^{(i)} - \widehat{\boldsymbol{b}}_{\text{MMSE}}(\tilde{\boldsymbol{b}}, \gamma_2)\|_2^2$    ▷ Compute error for estimated interference
        end for
        $\mathcal{L} = \frac{1}{B}\sum_{j=0}^{B-1}(\mathcal{L}_{s,j} + \mathcal{L}_{b,j})$    ▷ Monte Carlo approximation of pointwise MMSE
        $\widehat{\boldsymbol{b}}^{(i+1)} \leftarrow \text{RMSprop}\left(\widehat{\boldsymbol{b}}^{(i)}, \nabla_{\boldsymbol{b}}\mathcal{L}\right)$    ▷ Update estimated interference using RMSprop optimizer
    end for
    $\widehat{\boldsymbol{s}}^{(N)} = \boldsymbol{y} - \kappa\,\widehat{\boldsymbol{b}}^{(N)}$    ▷ Compute the estimated SOI
    return $\widehat{\boldsymbol{s}}^{(N)}, \widehat{\boldsymbol{b}}^{(N)}$
end function
```

For implementation, we consider $B = 96$ and $M = 800$, with learning rate $\eta$ as a cosine annealing rate with an initial value of 0.0005.

The performance of Algorithm 1 is evaluated based on MSE and BER of the signal estimate across 6 examples per SIR level, as shown in Fig. 7-1. Unfortunately, the exten-

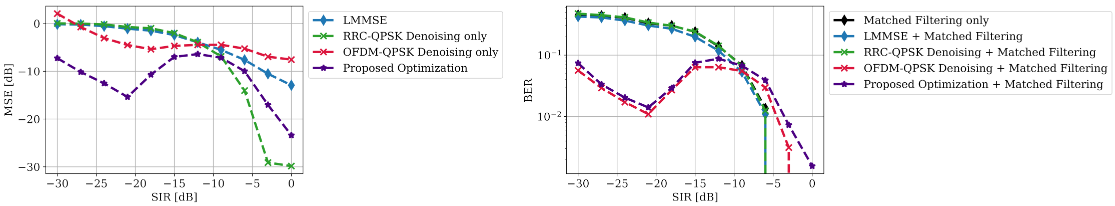


Figure 7-1: Comparing the MSE and BER of the estimated SOI from Algorithm 1 against traditional linear processing methods and against using individual MMSE denoisers only. (Note that each datapoint is an average of 6 test examples per SIR level, due to the algorithm's extensive runtime.)

sive runtime of the algorithm limits the evaluation to a small number of test examples. Nonetheless, we still hope to extract some preliminary insights from this restricted test set. We compare the performance of Algorithm 1 to traditional linear processing (i.e., LMMSE estimator and Matched Filtering without interference mitigation), as well as using the individual MMSE denoisers to estimate each signal component independently while treating the other component as AWGN.

When only the SOI denoiser is used, the resulting MSE and the BER of the estimated SOI are similar to those from linear processing methods. This reflects the method's suboptimal performance since the structures of the interference are not fully exploited in this approach.

On the other hand, we can also use the interference denoiser only. In this case, we first obtain a denoised estimate of the interference present by seeking, for a given $\kappa$,

$$\widehat{\boldsymbol{b}} = \widehat{\boldsymbol{b}}_{\mathrm{MMSE}}(\boldsymbol{y}/\kappa, \kappa^2)$$

and subsequently, estimate the SOI by

$$\widehat{\boldsymbol{s}} = \boldsymbol{y} - \kappa\widehat{\boldsymbol{b}}.$$

By denoising the interference, we can obtain a good estimate of the signal components, especially in low SIR regimes where the interference dominates. This is reflected in the improved recovery of the SOI bits, compared to matched filtering (which treats the interference as white noise). However, the MSE for this approach still underperforms compared to linear MMSE methods, since information pertaining to the SOI waveform structures is not utilized effectively (since only the interference denoiser model is used).

By using Algorithm 1, and initialized based on the denoised interference component, we fit the SOI and interference jointly. This approach hence leads to a better SOI estimate in the MSE sense while maintaining similar BER performance to the latter approach.

Nevertheless, we notice that the proposed algorithm falls short in the higher SIR regimes, and performs worse than linear methods in estimating the SOI at certain SIR levels. There are several shortcomings of Algorithm 1. For instance, obtaining a precise approximation of the objective function at each time step via Monte Carlo methods requires a large number of Gaussian noise realizations (corresponding to $B$) spread over a large integration support (corresponding to the limits of the uniform distribution which we draw $\gamma_1$ and $\gamma_2$). In Algorithm 1, we select these values to be relatively small due to runtime considering; but even so, the algorithm's runtime remains substantial, taking around 55 minutes per test signal (as opposed to U-Net's average of less than 1 second for separation). To enhance this method's effectiveness, better numerical integration and approximation strategies should be deployed in computing the objective function, especially given a limited number of samples per iteration.

## 7.3 Discussion and Concluding Remarks

This chapter offers a preliminary exploration of an alternative perspective in using MMSE estimators for source separation—specifically, the potential of independent MMSE denoisers as proxies for their source models in signal separation tasks. The theoretical foundation for this stems from the I-MMSE relation, which connects data distributions to optimal MMSE estimation in the Gaussian channel setting. While we can demonstrate its relevance in analytically tractable configurations such as with Gaussian sources, the practicality and effectiveness of this method in a data-driven approach warrant further investigation.

One significant challenge arises when moving beyond the Gaussian case, as an analytical form of the MMSE estimator is no longer easily derived. Although we propose the use of deep learning as function approximators to these MMSE denoisers, it remains uncertain how to evaluate the effectiveness of such neural networks in capturing the characteristics of the source models.

Another critical consideration and challenge is the inference step, which requires integration over an extensive range of SNR values. Our current approach approximates this through Monte Carlo methods and finds the minimizer via stochastic gradient descent. Yet, optimization within this potentially nonconvex landscape remains difficult, indicating that further algorithmic developments are needed in this context.

Meanwhile, research on denoising diffusion models for inverse problems has arrived at structurally similar algorithmic solutions, albeit from different premises and with differing theoretical justifications. It would be enlightening to uncover connections between these various perspectives. Notably, related work has demonstrated the effectiveness of a similar approach, i.e., using trained models as priors for separating digital communication waveforms [78]. The algorithm uses diffusion denoising models, and is conceptually based on a generalized $\alpha$-posterior based MAP estimator across different levels of Gaussian smoothing levels. However, this approach diverges from the I-MMSE approach in several significant ways, especially in the scaling of loss terms and the computation of the gradient updates. Future work entails establishing a clearer association between these two approaches and identifying and justifying key distinctions that these two approaches might have.

# Chapter 8

# Conclusion

In this thesis, we studied the problem of single-channel source separation, with a particular emphasis on RF systems using data-driven machine-learning methods. In our exposition, we formalize the complexities of the problem and study different regimes for their feasibility and inherent challenges.

A central theme in our work is the reexamination of traditional model-based approaches (with source models provided through a genie, thereby serving as a performance bound/baseline) to gain insights into the solution structures and compare how well our proposed data-driven methods ("blind" to the underlying signal models) perform. Specifically, we looked into two problem abstractions, which serve as instructive evaluations and benchmarks for our proposed deep-learning approaches. One, through a simplified prototype problem involving OFDM structures, we highlight the limitations of current deep-learning solutions (primarily designed for separating audio signals) and suggest appropriate modifications to the neural architectures for enhanced performance in the RF counterpart. Two, extending this exploration, we analyzed the impact of time shifts in the cyclostationary Gaussian time series on formulating an optimal estimator, providing a performance lower bound for comparison with novel data-driven methods. Through these scenarios, we seek to bridge the gap between data-driven methods and optimal model-based approaches in analytically tractable cases, and to demonstrate how the former can serve as attractive solutions in scenarios when the latter fails to be practicable.

On the empirical front, we also discuss the impact of neural architectural and hyper-parameter choices on the performance of deep learning solutions in tackling single-channel source separation problems. Equipped with these insights, we evaluate the proposed methods against real-world RF waveforms, setting their performance as the benchmark for the "RFChallenge", which we devised to help with the gap in the literature for a good comparative platform.

From an alternative perspective, we also explore the possibility of using insights about MMSE estimators to learn optimal denoisers for individual signal types. Leveraging recent results that relate such denoisers to data distributions, we investigate the utility of applying these individually trained denoising models for signal separation. This approach offers an alternative approach to the signal separation problem and is believed to present a more scalable strategy. Nonetheless, the practicality of such methods via a data-driven approach requires further exploration to reach competitive signal separation performance.

## 8.1 Challenges and Opportunities

This thesis delves into the capabilities of data-driven methods for single-channel source separation in RF systems, approached from different perspectives. However, this RF signal separation problem remains far from a solved puzzle. Several challenges persist, ranging from the optimization of neural architectures tailored for diverse RF conditions to designing solutions that seamlessly integrate our proposed data-driven techniques into RF systems. These challenges pave the way for future exploration opportunities. Specifically, we spotlight three major aspects to expand upon the work presented here.

**Adapting to the Dynamic Growth of Wireless Ecosystem:** As we witness innovations in system designs of next-generation wireless devices, including novel protocols and waveform designs, the challenge lies in crafting solutions adaptable to this growth and diversity. Historically, hand-engineered approaches served us well. But their practicality in this rapidly evolving landscape is questionable. As newer, potentially more intricate protocols emerge, how do we ensure our solutions remain scalable and robust? It is evident that effective and robust solutions, underpinned by a thorough understanding of their efficacy across diverse conditions, are paramount for future readiness.

**Separating Beyond Two-Component Mixtures:** Our exploration in this thesis predominantly centers around two-component mixtures, from which we have gleaned invaluable insights into such configurations. Yet, as the RF spectrum becomes increasingly crowded, we encounter scenarios with multiple (more than two) sources, complicating our problem at hand. On the one hand, we remark that our current methodologies can still be relevant, particularly by separating a reference signal (SOI) from all other components ("signal(s)-not-of-interest"), and iterating this process to separate the latter further. Nevertheless, such solutions might be computationally prohibitive. We recognize that addressing this more complex scenario may require novel strategies, and that what works for two components may not necessarily be suitable for multi-source mixtures—thereby presenting an opportunity for future research. For example, in our penultimate chapter, we made initial strides into scalable solutions by proposing to learn individual priors; the crux lies in addressing the core differences and technical challenges arising from such multi-source separation problem.

**Exploiting More Complex Structures in RF Signals:** Implicit to our problem abstractions we studied the focus on temporal structures tied to physical-layer communication properties, such as the symbol constellation and pulse-shaping functions to name a few. Yet, on a longer time scale, RF signals exhibit more intricate structures. These arise from different coding or packet-level structures, and distinct activity patterns associated with their application use cases. Leveraging these features more effectively could facilitate enhanced separation performances. Delving deeper into these long-term structures and sophisticated characteristics presents a promising direction for future research.

## 8.2 Bridging Model-Based and Data-Driven Approaches

More broadly, when viewed in conjunction with the collection of works on machine learning for RF systems discussed in Chapter 1, a compelling narrative emerges about the potential role of data-driven methods (and particularly deep learning techniques) in the design of future wireless systems. This thesis also highlights a critical aspect—the pitfalls of applying deep learning tools without careful consideration of the nuanced intricacies inherent to RF signals (e.g., as characterized in Chapter 3). Recalling the skepticism on the benefits of deep learning in this field, especially when comparing against engineering solutions tapping into decades of rich domain knowledge and expertise, our research presents a strong case for deep learning. We posit its advantages in scenarios where traditional, model-based approaches might be less feasible (e.g., in the absence of source models, and when desiring to go beyond linear model parameterizations). However, we also emphasize that a deep dive into model-based solutions has offered invaluable insights, aiding in the selection and design of the architecture for our learning-based approaches. The challenge—and indeed, the opportunity—is in effectively harnessing these data-driven methodologies, melded with judicious strategies that we can draw from model-based insights.

## 8.3 Concluding Remarks

We hope this research fosters growth in machine-learning approaches within the realm of RF source separation, paving the way for future studies that build on our findings and explore new neural architectures and algorithms that may present better gains. As an enabling tool, the proposed "RFChallenge" provides a platform for a methodical comparison and evaluation of these novel techniques. Ultimately, our aspiration is for our work to ignite further exploration at the intersection of machine learning and RF system design, fostering a new wave of innovations that push the boundaries of what is possible in next-generation wireless communication and sensing technology.

# Bibliography

[1] Zhongqiang Luo, Chengjie Li, and Lidong Zhu. A comprehensive survey on blind source separation for wireless adaptive processing: Principles, perspectives, challenges and new research directions. *IEEE Access*, 6:66685–66708, 2018.

[2] Taiwo Oyedare, Vijay K Shah, Daniel J Jakubisin, and Jeffrey H Reed. Interference suppression using deep learning: Current approaches and open challenges. *IEEE Access*, 2022.

[3] Hermann von Helmholtz. *On the sensations of tone as a physiological basis for the theory of music.* Longmans, Green, and Co., London, 1875.

[4] E. Colin Cherry. Some experiments on the recognition of speech, with one and with two ears. *The Journal of the Acoustical Society of America*, 25(5):975–979, 1953.

[5] E. Colin Cherry. *On Human Communication: A Review, a Survey, and a Criticism.* MIT Press Classics. MIT Press, 1980.

[6] Trausti Kristjansson, John Hershey, Peder Olsen, Steven Rennie, and Ramesh Gopinath. Super-human multi-talker speech recognition: The IBM 2006 speech separation challenge system. In *Ninth International Conference on Spoken Language Processing*, 2006.

[7] Yan-min Qian, Chao Weng, Xuan-kai Chang, Shuai Wang, and Dong Yu. Past review, current progress, and challenges ahead on the cocktail party problem. *Frontiers of Information Technology & Electronic Engineering*, 19(1):40–63, 2018.

[8] Joel Akeret, Chihway Chang, Aurelien Lucchi, and Alexandre Refregier. Radio frequency interference mitigation using deep convolutional neural networks. *Astronomy and computing*, 18:35–39, 2017.

[9] Nir Shlezinger and Ron Dabora. Frequency-shift filtering for OFDM signal recovery in narrowband power line communications. *IEEE Trans. Commun.*, 62(4):1283–1295, 2014.

[10] Mengchen Zhao, Xiujuan Yao, Jing Wang, Yi Yan, Xiang Gao, and Yanan Fan. Single-channel blind source separation of spatial aliasing signal based on stacked-LSTM. *Sensors*, 21(14), 2021.

[11] Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

[12] Pierre Comon and Christian Jutten. *Handbook of Blind Source Separation: Independent component analysis and applications.* Academic press, 2010.

[13] Laurent Benaroya, Frédéric Bimbot, and Rémi Gribonval. Audio source separation with a single sensor. *IEEE Trans. on Audio, Speech, and Language Process.*, 14(1):191–199, 2005.

[14] Emad M Grais, Mehmet Umut Sen, and Hakan Erdogan. Deep neural networks for single channel source separation. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pages 3734–3738. IEEE, 2014.

[15] Raphael Blouet, Guy Rapaport, Israel Cohen, and Cedric Fevotte. Evaluation of several strategies for single sensor speech/music separation. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pages 37–40. IEEE, 2008.

[16] Tu Shilong, Chen Shaohe, Zheng Hui, and Wan Jian. Particle filtering based single-channel blind separation of co-frequency MPSK signals. In *2007 International Symposium on Intelligent Signal Processing and Communication Systems*, pages 582–585, 2007.

[17] Tu Shilong, Zheng Hui, and Gu Na. Single-channel blind separation of two QPSK signals using per-survivor processing. In *APCCAS 2008 - 2008 IEEE Asia Pac. Conf. Circuits Syst.*, pages 473–476, 2008.

[18] Moeness G. Amin. Interference mitigation in spread spectrum communication systems using time-frequency distributions. *IEEE Trans. Signal Process.*, 45(1):90–101, 1997.

[19] Moeness G. Amin, Daniele Borio, Yimin D. Zhang, and Lorenzo Galleani. Time-frequency analysis for GNSSs: From interference mitigation to system monitoring. *IEEE Signal Process. Mag.*, 34(5):85–95, 2017.

[20] Timothy O'Shea and Nathan West. Radio machine learning dataset generation with GNU radio. *Proc. of GNU Radio Conf.*, 1(1), 2016.

[21] Timothy O'Shea, Tamoghna Roy, and T. Charles Clancy. Over-the-air deep learning based radio signal classification. *IEEE J. Sel. Topics Signal Process.*, 12(1):168–179, 2018.

[22] MIT RLE. RF Challenge - AI Accelerator. Accessed 2021-11-01.

[23] Daniel Stoller, Sebastian Ewert, and Simon Dixon. Wave-U-Net: A multi-scale neural network for end-to-end audio source separation. In *Proc. 19th International Society for Music Information Retrieval Conference*, pages 334–340, 2018.

[24] Aditya Arie Nugraha, Antoine Liutkus, and Emmanuel Vincent. Multichannel audio source separation with deep neural networks. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 24(9):1652–1664, 2016.

[25] Yosef Gandelsman, Assaf Shocher, and Michal Irani. "Double-DIP": Unsupervised image decomposition via coupled deep-image-priors. In *Proc. of IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 11026–11035, 2019.

[26] Youwei Lyu, Zhaopeng Cui, Si Li, Marc Pollefeys, and Boxin Shi. Reflection separation using a pair of unpolarized and polarized images. *Adv. Neural Inf. Process. Syst.*, 32:14559–14569, 2019.

[27] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Singing-voice separation from monaural recordings using deep recurrent neural networks. In *Proc. 15th International Society for Music Information Retrieval Conference*, pages 477–482, 2014.

[28] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Joint optimization of masks and deep recurrent neural networks for monaural source separation. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 23(12):2136–2147, 2015.

[29] Andreas Jansson, Eric J. Humphrey, Nicola Montecchio, Rachel M. Bittner, Aparna Kumar, and Tillman Weyde. Singing voice separation with deep U-Net convolutional networks. In *Proc. 18th International Society for Music Information Retrieval Conference*, pages 745–751, 2017.

[30] Yi Luo and Nima Mesgarani. Tasnet: Time-domain audio separation network for real-time, single-channel speech separation. In *Proc. of ICASSP*, pages 696–700, 2018.

[31] Yi Luo and Nima Mesgarani. Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 27(8):1256–1266, 2019.

[32] Efthymios Tzinis, Zhepei Wang, and Paris Smaragdis. Sudo RM-RF: Efficient networks for universal audio source separation. In *Proc. of MLSP*, pages 1–6, 2020.

[33] Yi Luo, Zhuo Chen, and Takuya Yoshioka. Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation. In *Proc. of ICASSP*, pages 46–50, 2020.

[34] Jingjing Chen, Qirong Mao, and Dong Liu. Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation. In *Proc. Interspeech 2020*, pages 2642–2646, 2020.

[35] Artemii Novoselov, Peter Balazs, and Götz Bokelmann. SEDENOSS: SEparating and DENOising seismic signals with dual-path recurrent neural network architecture. *J. Geophys. Res. Solid Earth*, 127(3):e2021JB023183, 2022.

[36] Mengchen Zhao, Xiujuan Yao, Jing Wang, Yi Yan, Xiang Gao, and Yanan Fan. Single-channel blind source separation of spatial aliasing signal based on stacked-LSTM. *Sensors*, 21(14), 2021.

[37] Xiaoqi Hou and Yong Gao. Single-channel blind separation of co-frequency signals based on convolutional network. *Digital Signal Processing*, 129:103654, 2022.

[38] Sven Hinderer. Blind source separation of radar signals in time domain using deep learning. In *2022 23rd International Radar Symposium (IRS)*, pages 486–491. IEEE, 2022.

[39] Hao Ma, Xiang Zheng, Lu Yu, Xingyu Zhou, and Yufan Chen. A novel end-to-end deep separation network based on attention mechanism for single channel blind separation in wireless communication. *IET Signal Processing*, 17(2):e12173, 2023.

[40] Gary CF Lee, Amir Weiss, Alejandro Lancho, Jennifer Tang, Yuheng Bu, Yury Polyanskiy, and Gregory W Wornell. Exploiting temporal structures of cyclostationary signals for data-driven single-channel source separation. In *2022 IEEE 32nd International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2022.

[41] Alejandro Lancho, Amir Weiss, Gary CF Lee, Jennifer Tang, Yuheng Bu, Yury Polyanskiy, and Gregory W Wornell. Data-driven blind synchronization and interference rejection for digital communication signals. In *GLOBECOM 2022-2022 IEEE Global Communications Conference*, pages 2296–2302. IEEE, 2022.

[42] Gary CF Lee, Amir Weiss, Alejandro Lancho, Yury Polyanskiy, and Gregory W Wornell. On neural architectures for deep learning-based source separation of co-channel ofdm signals. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[43] Timothy O'Shea and Jakob Hoydis. An introduction to deep learning for the physical layer. *IEEE Transactions on Cognitive Communications and Networking*, 3(4):563–575, 2017.

[44] Malte Schmidt, Dimitri Block, and Uwe Meier. Wireless interference identification with convolutional neural networks. In *2017 IEEE 15th International Conference on Industrial Informatics (INDIN)*, pages 180–185, 2017.

[45] Khalid Youssef, Louis Bouchard, Karen Haigh, Jan Silovsky, Bishal Thapa, and Chris Vander Valk. Machine learning approach to RF transmitter identification. *IEEE Journal of Radio Frequency Identification*, 2(4):197–205, 2018.

[46] Sebastian Dörner, Sebastian Cammerer, Jakob Hoydis, and Stephan Ten Brink. Deep learning based communication over the air. *IEEE Journal of Selected Topics in Signal Processing*, 12(1):132–143, 2017.

[47] Alexander Felix, Sebastian Cammerer, Sebastian Dörner, Jakob Hoydis, and Stephan Ten Brink. OFDM-autoencoder for end-to-end learning of communications systems. In *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5. IEEE, 2018.

[48] Yu Zhang, Muhammad Alrabeiah, and Ahmed Alkhateeb. Reinforcement learning of beam codebooks in millimeter wave and terahertz MIMO systems. *IEEE Transactions on Communications*, 70(2):904–919, 2021.

[49] George Turin. An introduction to matched filters. *IRE transactions on Information theory*, 6(3):311–329, 1960.

[50] Robert W Heath Jr. *Introduction to wireless digital communication: a signal processing perspective.* Prentice Hall, 2017.

[51] Norbert Wiener. *Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications.* MIT press Cambridge, MA, 1949.

[52] Paulo SR Diniz. *Adaptive filtering.* Springer, 1997.

[53] Han Li, Kean Chen, Lei Wang, Jianben Liu, Baoquan Wan, and Bing Zhou. Sound source separation mechanisms of different deep networks explained from the perspective of auditory perception. *Applied Sciences*, 12(2), 2022.

[54] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI 2015*, pages 234–241. Springer, 2015.

[55] Ivo Bizon Franco De Almeida, Luciano Leonel Mendes, Joel JPC Rodrigues, and Mauro AA Da Cruz. 5G waveforms for IoT applications. *IEEE Communications Surveys & Tutorials*, 21(3):2554–2567, 2019.

[56] Alan J. Coulson. Maximum likelihood synchronization for OFDM using a pilot symbol: algorithms. *IEEE Journal on Selected Areas in Communications*, 19(12):2486–2494, 2001.

[57] Bo Ai, Zhi-xing Yang, Chang-yong Pan, Jian-hua Ge, Yong Wang, and Zhen Lu. On the synchronization techniques for wireless OFDM systems. *IEEE Transactions on Broadcasting*, 52(2):236–244, 2006.

[58] Tevfik Yucek and Huseyin Arslan. OFDM signal identification and transmission parameter estimation for cognitive radio applications. In *IEEE GLOBECOM 2007 - IEEE Global Telecommunications Conference*, pages 4056–4060, 2007.

[59] Miao Shi, Y. Bar-Ness, and Wei Su. Blind OFDM systems parameters estimation for software defined radio. In *2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pages 119–122, 2007.

[60] Fayçal Ait Aoudia and Jakob Hoydis. End-to-end learning for ofdm: From neural receivers to pilotless communication. *IEEE Trans. on Wireless Comm.*, 21(2):1049–1063, 2022.

[61] Stefan Brennsteiner, Tughrul Arslan, John Thompson, and Andrew McCormick. A real-time deep learning OFDM receiver. *ACM Trans. Reconfigurable Technol. Syst.*, 15(3), dec 2022.

[62] *IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput*, 2009.

[63] Manuel Pariente, Samuele Cornell, Joris Cosentino, Sunit Sivasankaran, Efthymios Tzinis, Jens Heitkaemper, Michel Olvera, Fabian-Robert Stöter, Mathieu Hu, Juan M. Martín-Doñas, David Ditter, Ariel Frank, Antoine Deleforge, and Emmanuel Vincent. Asteroid: the PyTorch-based audio source separation toolkit for researchers. In *Proc. Interspeech*, 2020.

[64] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. *Advances in neural information processing systems*, 29, 2016.

[65] Anjana Punchihewa, Qiyun Zhang, Octavia A Dobre, C Spooner, Sreeraman Rajan, and R Inkol. On the cyclostationarity of OFDM and single carrier linearly digitally modulated signals in time dispersive channels: Theoretical developments and application. *IEEE Trans. Wirel. Commun.*, 9(8):2588–2599, 2010.

[66] Georgios B Giannakis. Cyclostationary signal analysis. In V. K. Madisetti and D. Williams, editors, *Digital Signal Processing Handbook*. CRC press, 1998.

[67] William A. Gardner. *Cyclostationarity in Communications and Signal Processing*. IEEE Press, 1994.

[68] William A Gardner. Cyclic wiener filtering: Theory and method. *IEEE Transactions on communications*, 41(1):151–163, 1993.

[69] William A. Gardner, Antonio Napolitano, and Luigi Paura. Cyclostationarity: Half a century of research. *Signal Processing*, 86:639–697, 4 2006.

[70] William A. Gardner. Common pitfalls in the application of stationary process theory to time-sampled and modulated signals. *IEEE Transactions on Communications*, 35(5):529–534, 1987.

[71] Bernard Picinbono and Pascal Chevalier. Widely linear estimation with complex data. *IEEE Trans. Signal Process.*, 43(8):2030–2033, 1995.

[72] François Chollet et al. Keras. https://keras.io, 2015.

[73] Martín Abadi et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. tensorflow.org.

[74] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. of 3rd Int. Conf. Learn. Represent.*, 2015.

[75] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

[76] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural Networks: Tricks of the Trade: Second Edition*, pages 437–478. Springer, 2012.

[77] Xianghao Kong, Rob Brekelmans, and Greg Ver Steeg. Information-theoretic diffusion. *arXiv preprint arXiv:2302.03792*, 2023.

[78] Tejas Jayashankar, Gary CF Lee, Alejandro Lancho, Amir Weiss, Yury Polyanskiy, and Gregory W Wornell. Score-based source separation with applications to digital communication signals. *arXiv preprint arXiv:2306.14411*, 2023.

# Endnotes

Part of the research work presented in this thesis was sponsored by the United States Air Force Research Laboratory and the Department of the Air Force Artificial Intelligence Accelerator and was accomplished under Cooperative Agreement Number FA8750-19-2-1000. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Department of the Air Force or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.