Title

Development of Machine Learning Algorithms for Low-Resolution MIMO Signal Processing

Author

Nguyen, Van Ly

Publication Date

2022

SAN DIEGO STATE UNIVERSITY
&
UNIVERSITY OF CALIFORNIA, IRVINE

Development of Machine Learning Algorithms for Low-Resolution MIMO Signal Processing

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computational Science

by

Van Ly Nguyen

Dissertation Committee:
Professor Duy H. N. Nguyen (SDSU), Advisor
Professor A. Lee Swindlehurst (UCI), Co-Advisor
Professor Ashkan Ashrafi (SDSU)
Professor Filippo Capolino (UCI)
Professor Ender Ayanoglu (UCI)

2022

DEDICATION

To my wife, my daughter, and my family!

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

LIST OF ACRONYMS

3GPP	Third Generation Partnership Project
ADC	Analog-to-Digital Converter
ADMM	Alternating Direction Method of Multipliers
AGC	Automatic Gain Control
Amp	Amplifier
AQNM	Additive Quantization Noise Model
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BMMSE	Bussgang decomposition-based Minimum Mean Squared Error
BPF	Band-Pass Filter
BS	Base Station
BWZF	Bussgang decomposition-based Weighted Zero Forcing
BZF	Bussgang decomposition-based Zero Forcing
CE-DD	Channel Estimation and Data Detection
CRC	Cyclic Redundancy Check
CSI	Channel State Information
CSIT	Channel State Information at Transmitter
dB	Decibel
DFT	Discrete Fourier Transform
DNN	Deep Neural Network
eMLD	empirical Maximum-Likelihood Detection
GAMP	Generalized Approximate Message Passing
IEEE	Institute of Electrical and Electronic Engineers
LDPC	Low-Density Parity-Check
LNA	Low-Noise Amplifier
LoS	Line-of-Sight
LPF	Low-Lass Filter
LS	Least-Squares
LTE	Long-Term Evolution
MAP	Maximum a Posteriori
MCD	Minimum-Center-Distance Detection
MIMO	Multiple-Input-Multiple-Output

ML	Maximum-Likelihood
MMD	Minimum-Mean-Distance Detection
MRC	Maximal Ratio Combining
NLoS	Non-Line-of-Sight
nML	near Maximum-Likelihood
NMSE	Normalized Mean Squared Error
NN	Nearest-Neighbor
OFDM	Orthogonal Frequency Division Multiplexing
OSD	One-bit Sphere Decoding
ReLU	Rectified Linear Unit
RF	Radio Frequency
RZF	Regularized Zero-Forcing
SC	Selective Combining
SIMO	Single-Input-Multiple-Output
SISO	Single-Input-Single-Output
SNR	Signal-to-Noise Ratio
SVM	Support Vector Machine
VB	Variational Bayes
VER	Vector Error Rate
ZF	Zero Forcing

NOTATION

- $\mathbb{R}$ and $\mathbb{C}$ denote the set of real and complex numbers, respectively, and j is the unit imaginary number satisfying $j^2 = -1$.
- Lower-case and upper-case boldface letters denote column vectors and matrices, respectively.
- $\mathbf{1}$ is a vector where every element is equal to one.
- $\mathbb{E}[\cdot]$ represents expectation.
- $\mathbb{P}[\cdot]$ is the probability of some event
- $\mathbb{I}[\cdot]$ represents the indicator function, which equals 1 if the argument event is true and equals 0 otherwise.
- Depending on the context, the operator $|\cdot|$ is used to denote the absolute value of a number, or the cardinality of a set.
- $\|\cdot\|$ and $\|\cdot\|_{\mathrm{F}}$ denote the ℓ_2-norm of a vector and the Frobenius norm of a matrix, respectively.
- The transpose and conjugate transpose are denoted by $[\cdot]^T$ and $[\cdot]^H$, respectively.
- The operator $\mathrm{mod}(a, b)$ calculates a modulo b.
- The notations $\mathrm{Var}[\cdot]$ and $\mathrm{Cov}[\cdot, \cdot]$ denote the variance and covariance, respectively.
- The integral $\Phi(a) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{a} e^{-t^2/2} dt$ is the cumulative distribution function (cdf) of the standard normal random variable.
- The notation $\Re\{\cdot\}$ and $\Im\{\cdot\}$ respectively denotes the real and imaginary parts of the complex argument.
- $\mathcal{N}(\cdot, \cdot)$ and $\mathcal{CN}(\cdot, \cdot)$ represent the real and the complex normal distributions respectively, where the first argument is the mean and the second argument is the variance or the covariance matrix.
- The operator $\mathrm{blkdiag}(\mathbf{A}_1, \ldots, \mathbf{A}_n)$ represents a block diagonal matrix, whose main-diagonal blocks are $\mathbf{A}_1, \ldots, \mathbf{A}_n$.

Note that if $\Re\{\cdot\}$, $\Im\{\cdot\}$, or $\Phi(\cdot)$ are applied to a matrix or vector, they are applied separately to every element of that matrix or vector.

ACKNOWLEDGMENTS

First, I would like to gratefully acknowledge and express my sincere thanks to my supervisor, Professor Duy H. N. Nguyen, and my co-supervisor, Professor A. Lee Swindlehurst, for their continuous guidance and support throughout my doctoral study. My special thanks go to Professor Duy H. N. Nguyen for giving me a wonderful opportunity to pursue my doctoral study at San Diego State University (SDSU) and the University of California, Irvine (UCI). His patience, timely comments, invaluable support, and guidance have certainly helped me complete this dissertation. My co-supervisor, Professor A. Lee Swindlehurst, has always been ready to give me advice and suggestions that I found important and helpful during the tough times of my doctoral study. It is a truly privilege for me to work under the supervision of Professor Duy H. N. Nguyen and Professor A. Lee Swindlehurst. I have learned so much from their remarkable technical knowledge and research enthusiasm. I would also like to express my gratitude to Professor Ashkan Ashrafi at SDSU, Professor Filippo Capolino and Professor Ender Ayanoglu at UCI, for serving in my Ph.D. committee, giving me insightful comments and raising many interesting questions in my defense.

I am deeply indebted to Professor Linh-Trung Nguyen for giving me enormous help and support during my undergraduate and master studies. I am grateful to Professor Pierre Duhamel, Professor Hikmet Sari, and Professor Jocelyn Fiorina for their support and guidance while I was studying for my master degree in France. I would also like to extend my warmest thanks to my friends Shivam Garg, Shuan He, Son Pham, Phu Tran, Duy Tran's family, Vu Tran's family, and my supervisor's family for their help and kindness, and for the wonderful time I have during my doctoral study.

I would not forget to gratefully acknowledge Computational Science Research Center (CSRC) of SDSU for running the computational science doctoral program that I am with. The work in this dissertation was supported in part by a startup fund from San Diego State University, in part by the teaching assistantship from the Department of Electrical and Computer Engineering, San Diego State University, and in part by the U.S. National Science Foundation under Grant CCF-1703635 and Grant ECCS-1824565.

Last but not least, I would like to express my deepest love and gratitude to my wife Mai who has always been with me, encouraging me, and giving me unwavering support. I am thankful for the joy and motivation I have from my lovely daughter An Vy. I also thank my parents and my brother Tam's family for always trusting in me and encouraging me to pursue my passion. I hope this accomplishment makes you all proud of me.

VITA

Van Ly Nguyen

EDUCATION

Doctor of Philosophy in Computational Science **2022**
University California, Irvine *Irvine, California*
San Diego State University *San Diego, California*

Master of Science in Wireless Communications Systems **2016**
CentraleSupélec, Paris-Saclay University *Gif-sur-Yvette, France*

Bachelor of Engineering in Electronics and Telecommunications **2014**
VNU – University of Engineering and Technology *Hanoi, Vietnam*

RESEARCH EXPERIENCE

Graduate Research Assistant **2017–2022**
San Diego State University *San Diego, California*

Intern **2016**
CentraleSupélec, Paris-Saclay University *Gif-sur-Yvette, France*

Research Assistant **2015**
VNU – University of Engineering and Technology *Hanoi, Vietnam*

TEACHING EXPERIENCE

Teaching Assistant:
EE458L – Communication and DSP Lab **2019–2022**
San Diego State University *San Diego, California*

REFEREED JOURNAL PUBLICATIONS

L. V. Nguyen, D. T. Ngo, N. H. Tran, A. L. Swindlehurst, and D. H. N. Nguyen, “Supervised and semi-supervised learning for MIMO blind detection with low-resolution ADCs,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2427–2442, Apr. 2020.

L. V. Nguyen, A. L. Swindlehurst, and D. H. N. Nguyen, “SVM-based channel estimation and data detection for one-bit massive MIMO systems,” *IEEE Trans. Signal Process.*, vol. 69, pp. 2086–2099, Mar. 2021.

Ly V. Nguyen, A. Lee Swindlehurst, and Duy H. N. Nguyen, “Linear and deep neural network-based receivers for massive MIMO systems with one-bit ADCs,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 11, pp. 7333–7345, Nov. 2021.

Ly V. Nguyen, Duy H. N. Nguyen, and A. Lee Swindlehurst, “Deep learning for estimation and pilot signal design in few-bit massive MIMO systems,” submitted to *IEEE Trans. Wireless Commun.* (under 2nd round review), 2022.

REFEREED CONFERENCE PUBLICATIONS

L. V. Nguyen, D. T. Ngo, N. H. Tran, and D. H. N. Nguyen, “Learning methods for MIMO blind detection with low-resolution ADCs,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, May 2018.

L. V. Nguyen, D. H. N. Nguyen, and A. L. Swindlehurst, “SVM-based channel estimation and data detection for massive MIMO systems with one-bit ADCs,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, Dublin, Ireland, June 2020.

Ly V. Nguyen, Duy H. N. Nguyen, and A. Lee Swindlehurst, “DNN-based detectors for massive MIMO systems with low-resolution ADCs,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, Montreal, Canada, June 2021.

ABSTRACT OF THE DISSERTATION

Development of Machine Learning Algorithms for Low-Resolution MIMO Signal Processing

By

Van Ly Nguyen

Doctor of Philosophy in Computational Science

San Diego State University and University of California, Irvine, 2022

Professor Duy H. N. Nguyen (SDSU), Chair

Due to the proliferation of mobile devices and services, the scale of multiple-input-multiple-output (MIMO) communication systems is getting larger and larger and can be massive in future wireless networks. This results in significant increases in hardware cost and power consumption. Recently, low-resolution analog-to-digital converters (ADCs) have been considered as a practical solution for reducing hardware cost and power consumption in MIMO systems. This is because low-resolution ADCs have simple hardware architectures as well as very low power consumption. However, the severe nonlinearity of low-resolution ADCs causes significant distortions in the received signals and therefore makes signal processing tasks such as channel estimation and data detection much more challenging compared to those in high-resolution systems. Motivated by the fact that machine learning is very powerful in solving non-linear problems, this dissertation exploits machine learning to develop low-complexity yet efficient and robust algorithms for channel estimation and data detection in MIMO systems with low-resolution ADCs.

First, the blind detection problem, i.e., detection without channel state information (CSI), in MIMO systems with low-resolution ADCs is studied. Two learning methods, which employ a sequence of pilot symbol vectors as the initial training data, are proposed. A performance analysis of the vector error rate is then derived for the case of 1-bit ADCs. Based on the analytical results, a criterion for designing transmitted signals is also presented.

Next, we show how support vector machine (SVM) – a well-known supervised-learning technique in machine learning – can be exploited to provide efficient and robust channel estimation and data detection in MIMO systems with 1-bit ADCs. Both uncorrelated and spatially correlated channels are considered. An SVM-based joint channel estimation and data detection method and an extension to frequency-selective fading channels will also be proposed.

Then, a deep learning framework for low-resolution MIMO channel estimation, data detection, and pilot signal design is proposed. The proposed estimation and detection networks are model-driven and have special structures that take advantage of domain knowledge in the low-resolution quantization process. An important feature of the proposed channel estimation network is that the pilot matrix is integrated into the weight matrices, which leads to a joint optimization of both the channel estimator at the base station and the pilot signal transmitted from the users. We also develop a nearest-neighbor search method to further improve the data detection performance.

Finally, via numerical results, the proposed machine learning-based methods are shown to be efficient and outperform existing ones. They are also shown to be robust against inherent computational issues in the low-resolution MIMO framework.

Chapter 1

Introduction

1.1 Wireless Communications

Wireless communication has a long history of over a hundred years dating back to the invention of the first photophone by Alexander Graham Bell and Charles Sumner Tainter in 1880. Nearly two decades later, in 1897, the first wireless telegraph system using radio waves was successfully developed by Guglielmo Marconi. However, the revolution of wireless communications did not really begin until the 1990s when the semiconductor technology achieved advanced developments with millions of electronic components packed in a single chip, and thus paved the way for the feasibility of implementing advanced digital signal processing techniques and algorithms. Since the start of the revolution, wireless communication technologies have quickly and fundamentally changed the way we live and communicate. Thanks to the over-the-air broadcast nature of electromagnetic waves, we can now keep connected wirelessly and almost anywhere were go. This mobility convenience has urged the rapid development of wireless applications and services. From only text and voice services in the early wireless generations, nowadays, we can use a wide variety of other high quality applications and services such as web browsing and transactions, on-demand multimedia streaming, online gaming, video conferencing, and so on, all possible through wireless links. We are even moving toward a society where smart homes, cities, and factories can be fully-connected. Wireless connectivity has become an essential part of our daily life.

With the proliferation of mobile devices and services, the amount of wireless data traffic has grown at an exponential pace for many years. The number of mobile users and connections is expected to reach 5.7 billion and 12.3 billion in 2022 [1], respectively; generating a mobile data traffic of about 77 billion gigabytes per month. The demand for ubiquitous access and very high-speed wireless links will definitely continue to increase in future wireless networks. Satisfying this demand is a very challenging problem due to two fundamental phenomena of wireless channels: *fading* and *interference*.

- Fading is the variation of the channel strength over time and frequency due to small-scale and large-scale effects [2]. The small-scale effect is the result of receiving multiple signal copies over multiple paths between the transmitter and receiver. Since the signal copies travel in different paths, they experience different attenuations, different delays, and different phase shifts. This results in constructive or destructive addition of the signal copies. While the constructive addition amplifies the received signal strength, the destructive addition causes attenuation. The large-scale effect is due to path loss via distance attenuation and shadowing by large objects such as buildings and hills. The fading effect causes channel uncertainty over time and frequency and therefore makes it very difficult to maintain reliable communication. In the worst case, the link can be completely disrupted if the channel is in *deep fade* (strong destructive addition of the signal copies). This deep fade phenomenon is highly probable in practice, especially when the transmitter and/or the receiver are moving. Furthermore, for reliable communication, the power of the received signal has to be above a certain minimum level since there always exists noise at the receiver. This requirement can be troublesome in practice due to the rapid attenuation over distance of electromagnetic waves, which is even more severe at high frequency bands, e.g., millimeter-wave (mmWave) bands.

- Interference is the phenomenon in that a wireless link is interfered by other co-channel links. It is a direct consequence of the broadcast nature of wireless channels and is therefore inevitable. Interference happens when different transmitters communicate with a common receiver, when a transmitter sends signals to unintended receivers, or when different transmitter–receiver pairs share a common channel. In cellular wireless networks, interference is often categorized into two sources: *intra-cell* interference and *inter-cell* interference. While the former interference source comes from devices in the same cell, the latter is from other cells. To mitigate inter-cell interference, a practical approach that has been used in commercial systems is called "*frequency reuse*". This approach assigns adjacent cells to operate on different bands and makes sure that the cells operating on the same band are far apart from each other. Since electromagnetic waves rapidly attenuate over distance, the inter-cell interference would be negligible and can be treated as background noise. The frequency reuse approach however reduces spectral efficiency since each cell can only exploit parts of the available spectrum. If universal frequency reuse (all cells operate on the entire available spectrum) is used, the spectral efficiency can be improved but advanced interference management and mitigation techniques are required.

Fading and interference are two critical aspects of wireless channels that have to be sufficiently addressed in order to achieve reliable communications. How to deal with fading and interference is considered as the central design of wireless communications systems [2].

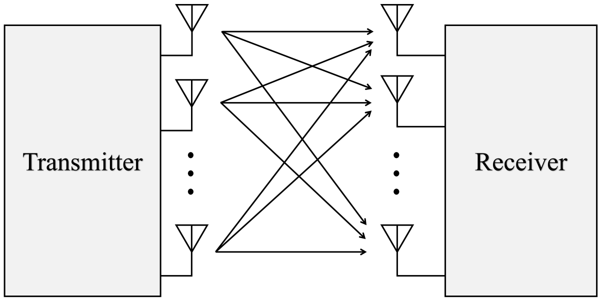


Figure 1.1: Illustration of a wireless communications system with multiple transmit and multiple receive antennas.

1.2 MIMO Communications: A Disruptive Wireless Technology

This section presents multiple-input multiple-output (MIMO) communications and briefly explains why it is a disruptive technology that can efficiently address the multi-path fading and interference issues as well as significantly improve the system throughput.

The term "multiple-input-multiple-output" originated from electric circuit and filter theory in the 1950s and referred to circuits that had multiple input and multiple output ports [3]. However, in the 1990s, it was adopted by the information and communication societies to refer to wireless communications systems that involve multiple antennas. Strictly speaking, MIMO indicates wireless systems equipped with multiple antennas at both the transmitter and receiver sides as illustrated in Figure 1.1. However, in a broader sense, it implies a collection of signal processing techniques that use multiple antennas to improve the performance of wireless communications systems and thus multiple antennas can be at the transmitter side, receiver side, or both.

A key feature of MIMO communications is the ability to combat multipath fading and improve the communication reliability through the creation of *diversity*, which refers to transmitting the same information over different wireless channels. Each channel in this context

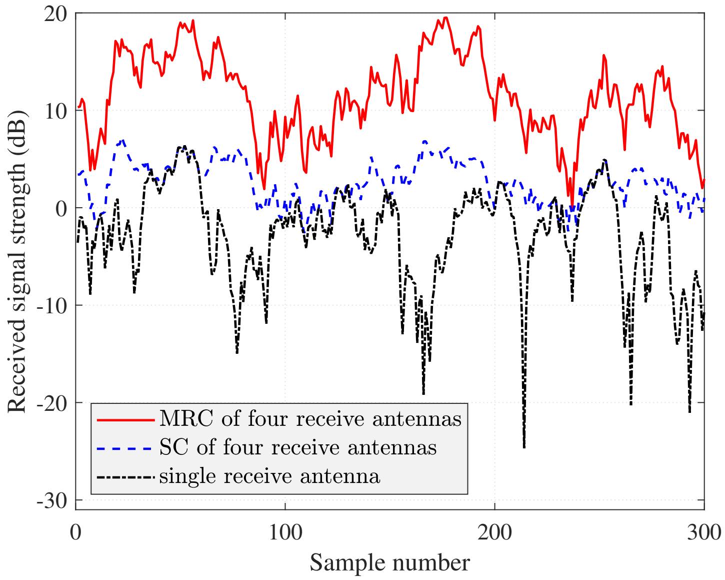


Figure 1.2: Simulated received signal strength comparison between one receive antenna and two combining methods SC and MRC of four receive antennas in Rayleigh fading with a single-antenna transmitter.

indicates the wireless link between a transmit and a receive antenna. Since these channels have different fading effects, the probability that all of them are in deep fade is significantly smaller than the case where only one receive antenna is used. Multiple observations of the transmitted signal obtained from multiple receive antennas are then used for a combining process where the transmitted information is recovered. The simplest combining method is called *selective combining* (SC), which simply takes the highest-power observation as the output of the combiner. An illustrative example for the benefits of diversity is given in Figure 1.2, which clearly shows that the fading issue can be efficiently addressed by the use of multiple antennas. There is a high chance that the channel is in deep fade if only one receive antenna is used. For systems with a single-antenna transmitter, the optimal combiner is called *maximal ratio combining* (MRC), whose output is a weighted sum of the observations from the receive antennas. The weights of an MRC are equal to the complex conjugate of the corresponding channel gains. In systems with a multiple-antenna transmitter, *space-time codes* are often used to improve the communication reliability. In space-time codes, redundant information for creating diversity is spread over both spatial and temporal dimensions.

Another key feature of MIMO communications is the ability to enhance system throughput through *spatial multiplexing*, which refers to simultaneously transmitting multiple data streams over a multipath fading environment without the need for increasing the transmission bandwidth. For example, each data stream is transmitted by one antenna of a multi-antenna transmitter. At the receiver side, if the number of receive antennas is greater than or equal to the number of transmit antennas, then a demultiplexing process can be utilized to recover the transmitted data streams. It was theoretically proved that the maximum number of data streams that can be supported by a MIMO system is given by the number of transmit antennas or the number of receive antennas, whichever is lesser [2, 3]. Thus, the system throughput can be significantly enhanced by increasing the number of deployed antennas.

The use of multiple antennas in MIMO communications can also helps mitigate interference. As mentioned earlier, interference is a direct consequence of the broadcast nature of wireless channels and is therefore inevitable. However, if the transmitted signal is directional instead of being broadcast over the air, interference can be mitigated. In the downlink transmission, the directivity of a transmitted signal can be obtained through a technique called *beamforming*. This technique controls the phase and amplitude of radiated waves emitted from the antenna elements so that the superposition of the waves forms a transmitted signal which travels along intended directions. This means the beamforming technique makes the radiated waves add constructively in the intended directions but destructively in the others. The signal part in each intended direction is set to carry data for a corresponding receiver. It is well known that the directional intensity of the transmitted signal is proportional to the number of transmit antennas [4] as illustrated in Figure 1.3. More transmit antennas result in more directional beamforming and consequently less interference. In cellular networks, beamforming helps reduce both intra-cell and inter-cell interferences. Another advantage of beamforming is that, since the transmitted power is focused on certain intended directions,

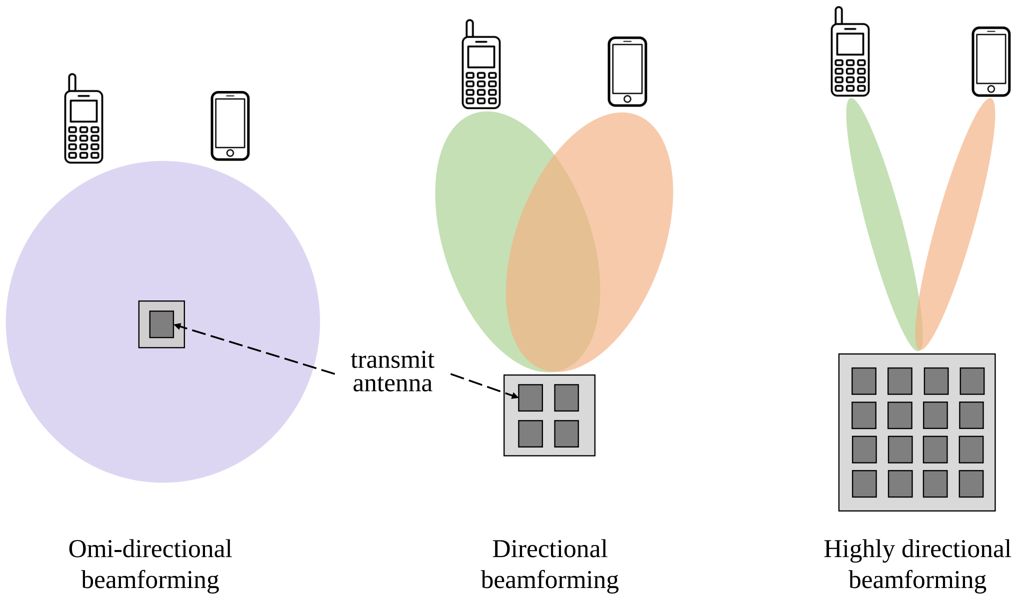
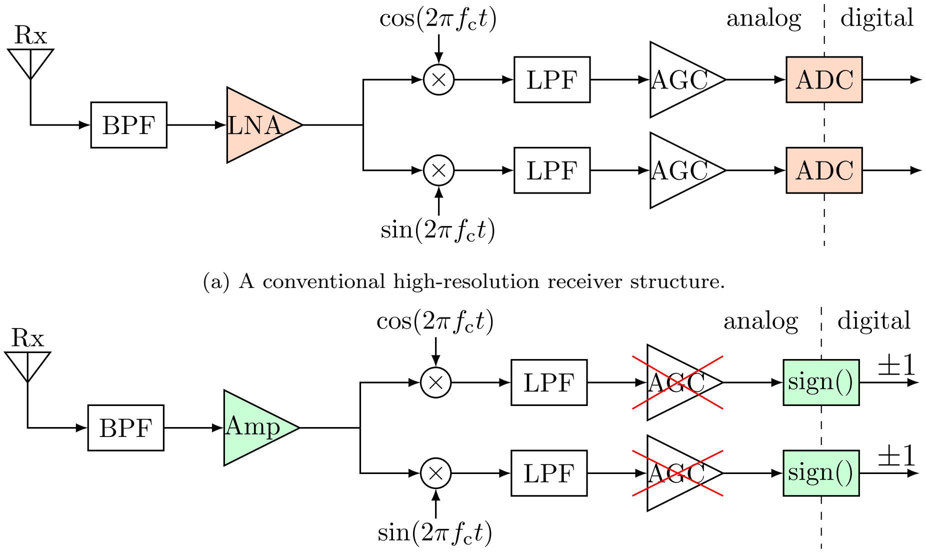


Figure 1.3: Directional beamforming through the use of multiple transmit antennas.

the transmitted signal can travel farther and so resulting in a larger coverage area. In the cellular uplink transmission, users are physically apart from each other and if they have no corporation, beamforming from them is not possible and interference at the base station certainly occurs. However, if the BS is equipped with multiple antennas and the number of BS antennas is greater than or equal to the number of data streams transmitted from the users, the BS can use observations from its multiple receive antennas to perform interference cancellation and extract data transmitted from different users. Such uplink interference cancellation is similar to demultiplexing as mentioned earlier.

1.3 Low-Resolution MIMO Communications

This section presents the research motivation for this dissertation. As explained in the previous section, MIMO is a disruptive wireless technology to improve the communication reliability, enhance system throughput, and mitigate interference. All these benefits are achievable at the expense of hardware cost and power consumption due to the use of multiple



(a) A conventional high-resolution receiver structure.

(b) A 1-bit ADC receiver structure.

Figure 1.4: Conventional high-resolution versus 1-bit receiver structures: (a) The ADCs have a high-resolution, e.g. 12-16 bits, and the LNA is required to behave linearly; (b) The 1-bit ADC is equivalent to a sign function, which can be implemented by a simple comparator. The AGC is not needed and the LNA can be replaced by a simple amplifier (Amp).

antennas. In particular, each receive antenna is connected to a radio frequency (RF) chain consisting of a series of components such as band-pass filter (BPF), low-noise amplifier (LNA), mixer, low-pass filter (LPF), automatic gain control (AGC), and analog-to-digital converters (ADC) as illustrated in Figure 1.4a. Conventionally, the RF chain is designed in a way that minimizes signal distortion and makes the overall system behavior linear. This asks for high-quality components in the RF chains, e.g., highly-linear LNAs and high-resolution ADCs. Thus, hardware cost and power consumption in MIMO systems are clearly much more critical compared to single-antenna systems.

Recently, massive MIMO has been widely considered as one of the core technologies for emerging 5G and future wireless networks [5–9]. This is because massive MIMO can improve the system performance by several orders of magnitude over small-scale MIMO systems thanks to the significant increase in the spatial degrees of freedom obtained by combining tens to hundreds of antennas at the BS [10–13]. This means the hardware cost and power consumption problems in massive MIMO systems are even more severe since scaling the conventional RF chain implementation in Figure 1.4a to a massive number of antennas can be too costly and power-consuming.

Since high-resolution ADCs are expensive and very power-hungry, e.g., the hardware complexity and power consumption of a flash ADC are exponentially proportional to the resolution bit [14, 15], a practical approach to the hardware cost and power consumption problems in MIMO systems is to use low-resolution (e.g., 1–3 bits) ADCs. The use of low-resolution ADCs not only reduces their hardware complexity but also results in the simplification or removal of other components in an RF chain. For example, as illustrated in Figure 1.4b, the architecture of a 1-bit ADC is as simple as a comparator whose power consumption is negligible. When one-bit ADCs are used, the AGC can be removed since only the *sign* of the real and imaginary parts of the received signals is retained. The stringent linearity requirement of the LNA can be relaxed and a simpler low-cost amplifier can be used instead. In addition, the use of low-resolution ADCs also helps reduce the prohibitive demand for high bandwidth on the fronthaul link between the baseband processing unit and the RF chains. For example, a receiver that is equipped with 500 antennas, where each antenna employs two separate ADCs for the in-phase and quadrature components, and where each ADC samples at a rate of 1 GS/s with 10-bit precision would produce 10 Terabit/s of data, which is much higher than the rates of the common public radio interface in today's fiber-optical fronthaul links [16].

The benefits on the hardware side make it easy to deploy low-resolution ADCs in practical systems. However, the lower-complexity and lower-power-consumption hardware necessitates special care in the subsequent signal processing. More specifically, the severe nonlinearities introduced by the low-resolution ADCs make signal processing tasks such as channel estimation and data detection in low-resolution MIMO systems much more challenging compared to those in high-resolution systems. Therefore, it is crucial that efficient signal processing methods for channel estimation and data detection be developed for such systems so that they can be transitioned to commercial systems.

1.4 Dissertation Contributions and Organization

This dissertation deals with the channel estimation and data detection problems in MIMO systems with low-resolution ADCs. The idea is to exploit machine learning to address the severe nonlinarities caused by the low-resolution ADCs since machine learning techniques have been shown to be powerful in solving nonlinear problems. The main contributions of this dissertation are to develop machine learning-based low-complexity yet efficient frameworks for channel estimation and data detection in low-resolution MIMO systems. We show via numerical results that the proposed solution approaches significantly outperform existing methods. Additionally, the developed algorithms are also robust against inherent computational issues in low-resolution MIMO systems. The remainder of the dissertation is organized as follows:

Chapter 2 starts with describing a general system model, a quantization model, and the problem of interest. Then, a literature survey of related works is given. Finally, the chapter presents the Bussgang decomposition from which several linear receivers are introduced.

Chapter 3 studies the blind detection problem in single-user MIMO systems with low-resolution ADCs. Blind detection in this context means detection without information about the channel state information (CSI). Two learning methods, which employ a sequence of pilot symbol vectors as the initial training data, are proposed. The first method exploits the use of a cyclic redundancy check (CRC) to obtain more training data, which helps improve the detection accuracy. The second method is based on the perspective that the to-be-decoded data can itself assist the learning process, so no further training information is required except the pilot sequence. For the extreme case of 1-bit ADCs, we provide a performance analysis of the vector error rate (VER) for the proposed methods. Based on the analytical results, a criterion for designing transmitted signals is also presented.

In Chapter 4, we show how *support-vector machine* (*SVM*), a well-known supervised-learning model in machine learning, can be exploited to provide efficient and robust channel estimation and data detection in massive MIMO systems with 1-bit ADCs. First, the problem of channel estimation for uncorrelated channels is formulated as a conventional SVM problem. The objective function of this SVM problem is then modified for estimating spatially correlated channels. Next, a two-stage detection algorithm is proposed where SVM is further exploited in the first stage. The performance of the proposed data detection method is very close to that of maximum-likelihood (ML) data detection when the channel is perfectly known. We also propose an SVM-based joint Channel Estimation and Data Detection (CE-DD) method, which makes use of both the to-be-decoded data vectors and the pilot data vectors to improve the estimation and detection performance. Finally, an extension of the proposed methods to OFDM systems with frequency-selective fading channels is presented.

In Chapter 5, we propose a deep learning framework for channel estimation, data detection, and pilot signal design to address the nonlinearity in low-resolution MIMO systems. The proposed channel estimation and data detection networks are model-driven and have special structures that take advantage of domain knowledge in the low-resolution quantization process. While the first data detection network, B-DetNet, is based on a linearized model obtained from the Bussgang decomposition, the channel estimation network and the second data detection network, FBM-CENet and FBM-DetNet respectively, rely on the original quantized system model. To develop FBM-CENet and FBM-DetNet, the maximum-likelihood channel estimation and data detection problems are reformulated to overcome the indeterminant gradient issue. An important feature of the proposed FBM-CENet structure is that the pilot matrix is integrated into the weight matrices of its channel estimator. Thus, training the proposed FBM-CENet enables a joint optimization of both the channel estimator at the base station and the pilot signal transmitted from the users. We also propose a nearest-neighbor search method to further improve the data detection performance. Unlike existing search methods that typically perform the search over a large candidate set, the proposed search method generates a limited number of most likely candidates and thus limits the search complexity.

Finally, Chapter 6 presents concluding remarks and potential directions for future work.

Chapter 2

Problem Statement and Literature Survey

2.1 Problem Statement

2.1.1 General System Model

We consider an uplink MIMO system where the transmitter side can be an N_{tx}-antenna user or N_{tx} single-antenna users that are located apart from each other and the receiver side is a base station (BS) equipped with N_{rx} receive antennas. It is assumed that $N_{\text{rx}} > N_{\text{tx}}$. Let $\mathbf{x}^{\mathbb{C}} \in \mathbb{C}^{N_{\text{tx}}}$ and $\mathbf{H}^{\mathbb{C}} \in \mathbb{C}^{N_{\text{rx}} \times N_{\text{tx}}}$ denote the transmitted signal vector and the channel matrix, respectively. In this dissertation, the superscript $^{\mathbb{C}}$ is used to indicate the complex domain. Unless otherwise stated, the channel is assumed to be block flat fading, i.e., it does not change over a certain interval of time. The unquantized received signal vector at the base station $\mathbf{r}^{\mathbb{C}} \in \mathbb{C}^{N_{\text{rx}}}$ is given as

$$\mathbf{r}^{\mathbb{C}} = \mathbf{H}^{\mathbb{C}}\mathbf{x}^{\mathbb{C}} + \mathbf{z}^{\mathbb{C}} \tag{2.1}$$

where $\mathbf{z}^{\mathbb{C}} \in \mathcal{CN}(\mathbf{0}, N_0\mathbf{I}_{N_{\text{rx}}})$ is a noise vector. Each received analog signal is quantized by a pair of b-bit ADCs, denoted as $\mathcal{Q}_b$, to produce the quantized received signal:

$$\mathbf{y}^{\mathbb{C}} = \mathcal{Q}_b\left(\mathbf{r}^{\mathbb{C}}\right) = \mathcal{Q}_b\left(\Re\{\mathbf{r}^{\mathbb{C}}\}\right) + j\mathcal{Q}_b\left(\Im\{\mathbf{r}^{\mathbb{C}}\}\right). \tag{2.2}$$

For vector or matrix arguments, the operator $\mathcal{Q}_b(\cdot)$ is applied separately to every element.

2.1.2 Quantization Model

The considered system employs an ADC that performs b-bit mid-rise uniform scalar quantization, $b \in \{1, 2, 3, \ldots\}$. The b-bit ADC model is characterized by a set of $2^b - 1$ thresholds denoted as $\{\tau_1, \ldots, \tau_{2^b-1}\}$. Without loss of generality, we assume $-\infty = \tau_0 < \tau_1 < \ldots < \tau_{2^b-1} < \tau_{2^b} = \infty$. Let Δ be the step size, so the thresholds of the uniform quantizer are given as

$$\tau_l = (-2^{b-1} + l)\Delta, \text{ for } l \in \mathcal{L} = \{1, \ldots, 2^b - 1\}. \tag{2.3}$$

The step size Δ is chosen to minimize the distortion between the quantized and non-quantized signals. The optimal value of Δ depends on the distribution of the input signals [17]. For standard Gaussian signals, the optimal step size $\Delta_{\text{opt}}^{\text{standard}}$ can be found numerically as in [18]. For non-standard complex Gaussian signals with variance $\sigma_{\text{rx}}^2 \neq 1$, the optimal step size for each real/imaginary signal component can be computed as $\Delta_{\text{opt}} = \sqrt{\sigma_{\text{rx}}^2/2}\Delta_{\text{opt}}^{\text{standard}}$. The quantized output is then defined as

$$\mathcal{Q}_b(r) = q_l = \begin{cases} \tau_l - \frac{\Delta}{2} & \text{if } r \in (\tau_{l-1}, \tau_l] \text{ with } l \in \mathcal{L} \\ (2^b - 1)\frac{\Delta}{2} & \text{if } r \in (\tau_{2^b-1}, \tau_{2^b}]. \end{cases} \tag{2.4}$$

For the case of 1-bit quantization, the ADC is equivalent to a sign$(\cdot)$ function, which means

$$\mathcal{Q}_1(r) = \text{sign}(r) = \begin{cases} +1 & \text{if } r \geq 0, \\ -1 & \text{otherwise.} \end{cases} \tag{2.5}$$

2.1.3 Problem of Interest

This dissertation focuses on the channel estimation and data detection in MIMO systems equipped with low-resolution ADCs. Each block fading interval of length T_b is divided into two phases. In the first phase, a pilot sequence $\mathbf{X}_\text{t}^\mathbb{C} \in \mathbb{C}^{N_\text{tx} \times T_\text{t}}$ of length T_t is used to generate the training data

$$\mathbf{Y}_\text{t}^\mathbb{C} = \mathcal{Q}_b\left(\mathbf{R}_\text{t}^\mathbb{C}\right) = \mathcal{Q}_b\left(\mathbf{H}^\mathbb{C}\mathbf{X}_\text{t}^\mathbb{C} + \mathbf{Z}_\text{t}^\mathbb{C}\right). \tag{2.6}$$

In the second phase, a data matrix $\mathbf{X}_\text{d}^\mathbb{C} \in \mathbb{C}^{N_\text{tx} \times T_\text{d}}$ where $T_\text{d} = T_\text{b} - T_\text{t}$ is transmitted and the received data matrix is given as

$$\mathbf{Y}_\text{d}^\mathbb{C} = \mathcal{Q}_b\left(\mathbf{R}_\text{d}^\mathbb{C}\right) = \mathcal{Q}_b\left(\mathbf{H}^\mathbb{C}\mathbf{X}_\text{d}^\mathbb{C} + \mathbf{Z}_\text{d}^\mathbb{C}\right) \tag{2.7}$$

The problem of interest is to estimate the channel matrix $\mathbf{H}^\mathbb{C}$ and detect the transmitted data matrix $\mathbf{X}_\text{d}^\mathbb{C}$ using $\mathbf{X}_\text{t}^\mathbb{C}$, $\mathbf{Y}_\text{t}^\mathbb{C}$, and $\mathbf{Y}_\text{d}^\mathbb{C}$. Note that the subscripts $_\text{t}$ and $_\text{d}$ are used to indicate the training and data transmission phases, respectively.

The work in this dissertation assumes perfect synchronization between the transmitter and receiver sides. It is also assumed that there is no inter-cell interference as single-cell communication is considered in this work. Additionally, we assume no hardware impairments in the transceiver.

2.2 Literature Survey

One of the first studies on MIMO systems with low-resolution ADCs is in [19], which shows that the mutual information of 1-bit ADC MIMO systems degraded by only a factor of $2/\pi$ at low SNRs compared to systems with infinite-resolution ADCs. Since then, a lot more attention and efforts have been spent on this research topic. The capacity in case of correlated noise and spatially correlated channels are studied in [20] and [21], respectively. Bounds on the high SNR capacity are derived in [22]. Capacity analysis with channel state information at transmitter (CSIT) is carried on in [23]. An approximate uplink achievable rate for massive MIMO systems is calculated in [24] by using the additive quantization noise model (AQNM). The achievable rate of hybrid analog-digital MIMO architectures is investigated in [25, 26]. A study of achievable rate for mixed-ADC massive MIMO systems is in [27], which is extended for frequency-selective channels in [28]. While a capacity lower bound for wideband massive MIMO systems with a large number of channel taps is derived in [29], throughput analysis based on the Bussgang decomposition is performed in [30].

Channel estimation for massive MIMO systems with low-resolution ADCs has attracted significant research interest and has been studied intensively. The majority of the existing approaches focus on one-bit systems in different scenarios, e.g., [31–51]. For example, maximum-likelihood (ML) and least-squares (LS) channel estimators were proposed in [31] and [32], respectively. The work in [33] exploits the Bussgang decomposition to form a one-bit Bussgang-based minimum mean-squablue error (BMMSE) channel estimator. Another BMMSE channel estimator was also proposed in [34] but for one-bit spatial sigma-delta ADCs in a spatially oversampled array. Channel estimation with temporally oversampled one-bit ADCs is studied in [35] and [36]. It has been shown that one-bit ADCs with spatial and temporal oversampling can help improve the channel estimation accuracy but more resources and computation are required due to the oversampling process. A channel estimation method based on SVM with 1-bit ADCs, referred to as soft-SVM, was presented in [37]. Angular-domain channel estimation for one-bit massive MIMO systems was studied in [38–40]. Spatially/temporally correlated channels and multi-cell processing with pilot contamination were investigated in [41] and [42], respectively. For sparse millimeter-wave MIMO channels, ML and maximum a posteriori (MAP) channel estimation were examined in [43] and [44], respectively. Taking into account the sparsity of such channels, the one-bit ADC channel estimation problem has been formulated as a compressed sensing problem in [45–47]. Performance bounds on the channel estimation of mmWave one-bit massive MIMO channels were reported in [48]. It is also worth noting that the work in [49] requires multiple OFDM symbols in the training sequence and the work in [51] is restricted to systems with only one single-antenna user.

Recently, there are several MIMO channel estimation methods for few-bit ADCs [52–57]. For example, the Bussgang decomposition was exploited in [52] to derive two linear channel estimators for few-bit ADCs including an extension of the BMMSE approach as well as a Bussgang-based weighted zero-forcing (BWZF) algorithm. A DNN-based joint pilot signal and channel estimator design is proposed in [53] where a conventional DNN structure was used. The work in [54, 55] studied mixed-resolution channel estimation where low-resolution ADCs were used for only some of the receive antennas, while the rest are equipped with conventional ADCs. The works in [56, 57] address the sparse channel estimation problem in massive MIMO systems where both hybrid analog-digital processing and low-resolution ADCs are utilized.

Data detection in MIMO systems with low-resolution ADCs has also been studied intensively in the literature, e.g., [31, 52, 58–72]. The one-bit ML detection problem is formulated in [31, 58]. For large-scale systems where ML detection is impractical, the authors in [31] proposed a so-called near-ML (nML) data detection method. The ML and nML methods are however non-robust at high signal-to-noise ratios (SNRs) when channel state information (CSI) is imperfectly known. ML detection with low-resolution ADCs is studied in [59, 60]. A one-bit sphere decoding (OSD) technique was proposed in [61]. However, the OSD technique requires a preprocessing stage whose computational complexity for each channel realization is exponentially proportional to both the number of receive and transmit antennas. The exponential computational complexity of OSD makes it difficult to implement in large scale MIMO systems. Generalized Approximate Message Passing (GAMP) and Bayes inference are exploited in [62, 63] but the proposed methods are sophisticated and expensive to implement. Various one-bit linear detectors were introduced in [52, 64] and several learning-based methods are also proposed in [65,66,68]. The linear receivers in [64] are easy to implement but their performance is often limited by an error floor. The learning-based methods in [65–67] are blind detection methods for which CSI is not required, but they are restricted to MIMO systems with a small number of transmit antennas and only low-dimensional constellations. A DNN-based one-bit detector was proposed in [73] but it requires online training since the network has to be retrained whenever the channel changes. This significantly increases the computational complexity and resources as well as the pilot overhead. Several other data detection approaches were proposed in [68, 70–72], but they are only applicable in systems where either a Cyclic Redundancy Check (CRC) [68, 70, 71] or an error correcting code such as Low-Density Parity-Check (LDPC) code [72] is available. The authors in [69] proposed a one-bit detection method based on the alternating direction method of multipliers (ADMM) algorithm that takes hardware impairments into account.

2.3 Bussgang Decomposition-based Linear Receivers

In the low-resolution MIMO literature, the Bussgang decomposition is often used for system analysis and algorithm development. The reason is that the Bussgang decomposition helps linearize a non-linear system. Thus, the Bussgang decomposition can be used to address the nonlinearities caused by the low-resolution ADCs. In this section, we introduce the Bussgang theorem [74] and the Bussgang decomposition, which is then used to derive linear channel estimators and data detectors.

2.3.1 Bussgang Decomposition

To obtain the Bussgang decomposition, we start with the Bussang theorem, which states that the cross-correlation between two Gaussian signals before and after one of them has passed through a nonlinear operation is the same, albeit a scaling factor. Theorem 2.1 below is for the case of one-dimensional signals.

Theorem 2.1 (One-dimensional Bussgang theorem [74])**.** *Consider two jointly circularly symmetric Gaussian random variables $r \in \mathbb{R}$ and $w \in \mathbb{R}$. Let $f_{\text{nl}} : \mathbb{R} \to \mathbb{R}$ be a non-linear distortion function. The cross-correlation of $y = f_{\text{nl}}(r)$ and w is $C_{yw} = VC_{rw}$ where V is called the Bussgang gain and given as $V = C_{yr}/C_r$.*

An extension to the case of multi-dimensional signals is given in the following Theorem.

Theorem 2.2 (Multi-dimensional Bussgang theorem [75])**.** *Consider the jointly circularly symmetric Gaussian random vectors $\mathbf{r} \in \mathbb{R}^M$ and $\mathbf{w} \in \mathbb{R}^M$. Let $\mathbf{y} = \mathbf{f}_{\text{nl}}(\mathbf{r})$ be the output of a non-linear distortion function where $\mathbf{f}_{\text{nl}} : \mathbb{R}^M \to \mathbb{R}^M$. The cross-correlations $\mathbf{C}_{\mathbf{yw}} = \mathbb{E}[\mathbf{y}\mathbf{w}^T]$ and $\mathbf{C}_{\mathbf{rw}} = \mathbb{E}[\mathbf{r}\mathbf{w}^T]$ are then related as $\mathbf{C}_{\mathbf{yw}} = \mathbf{C}_{\mathbf{yr}}\mathbf{C}_{\mathbf{r}}^{-1}\mathbf{C}_{\mathbf{rw}}$.*

A direct consequence of Theorem 2.2 is the Bussgang decomposition, which is given as

$$\mathbf{y} = \mathbf{f}_{\text{nl}}(\mathbf{r}) = \mathbf{V}\mathbf{r} + \boldsymbol{\eta} \tag{2.8}$$

where $\mathbf{V} = \mathbf{C}_{\mathbf{yr}}\mathbf{C}_{\mathbf{r}}^{-1}$ and the additive distortion term $\boldsymbol{\eta}$ is uncorrelated with $\mathbf{r}$. In general, $\boldsymbol{\eta}$ is not Gaussian, but it is often assumed in the literature to be Gaussian for ease of derivations.

2.3.2 Bussgang Decomposition-based Linear Channel Estimators

Here we consider the channel estimation task, which is done in the training phase. We start with vectorizing the received signal in (2.6) to obtain

$$\mathbf{y}_{\mathrm{t}}^{\mathbb{C}} = \mathcal{Q}_b(\mathbf{P}^{\mathbb{C}}\mathbf{h}^{\mathbb{C}} + \mathbf{z}_{\mathrm{t}}^{\mathbb{C}}), \tag{2.9}$$

where $\mathbf{y}_{\mathrm{t}}^{\mathbb{C}} = \mathrm{vec}(\mathbf{Y}_{\mathrm{t}}^{\mathbb{C}}) \in \mathbb{C}^{N_{\mathrm{rx}}T_{\mathrm{t}}}$, $\mathbf{P}^{\mathbb{C}} = \mathbf{X}_{\mathrm{t}}^{T} \otimes \mathbf{I}_{N_{\mathrm{rx}}} \in \mathbb{C}^{N_{\mathrm{rx}}T_{\mathrm{t}} \times N_{\mathrm{rx}}N_{\mathrm{tx}}}$, $\mathbf{h}^{\mathbb{C}} = \mathrm{vec}(\mathbf{H}^{\mathbb{C}}) \in \mathbb{C}^{N_{\mathrm{rx}}N_{\mathrm{tx}}}$, and $\mathbf{z}_{\mathrm{t}}^{\mathbb{C}} = \mathrm{vec}(\mathbf{Z}_{\mathrm{t}}^{\mathbb{C}}) \in \mathbb{C}^{N_{\mathrm{rx}}T_{\mathrm{t}}}$. We convert the notation in (2.9) into the real domain as

$$\mathbf{y}_{\mathrm{t}} = \mathcal{Q}_b(\mathbf{P}\mathbf{h} + \mathbf{z}_{\mathrm{t}}) \tag{2.10}$$

where

$$\mathbf{y}_{\mathrm{t}} = \begin{bmatrix} \Re\{\mathbf{y}_{\mathrm{t}}^{\mathbb{C}}\} \\ \Im\{\mathbf{y}_{\mathrm{t}}^{\mathbb{C}}\} \end{bmatrix}, \ \mathbf{h} = \begin{bmatrix} \Re\{\mathbf{h}^{\mathbb{C}}\} \\ \Im\{\mathbf{h}^{\mathbb{C}}\} \end{bmatrix}, \text{ and } \mathbf{P} = \begin{bmatrix} \Re\{\mathbf{P}^{\mathbb{C}}\} & -\Im\{\mathbf{P}^{\mathbb{C}}\} \\ \Im\{\mathbf{P}^{\mathbb{C}}\} & \Re\{\mathbf{P}^{\mathbb{C}}\} \end{bmatrix}.$$

Note that $\mathbf{y}_{\mathrm{t}} \in \mathbb{R}^{2N_{\mathrm{rx}}T_{\mathrm{t}}}$, $\mathbf{h} \in \mathbb{R}^{2N_{\mathrm{rx}}N_{\mathrm{tx}}}$, $\mathbf{P} \in \mathbb{R}^{2N_{\mathrm{rx}}T_{\mathrm{t}} \times 2N_{\mathrm{rx}}N_{\mathrm{tx}}}$, and $\mathbf{z}_{\mathrm{t}} \in \mathbb{R}^{2N_{\mathrm{rx}}T_{\mathrm{t}}}$. We now apply the Bussgang decomposition to (2.10) to obtain the Bussgang decomposition-based linear channel estimators BMMSE and BWZF for low-resolution massive MIMO systems [33, 52]. The system model in (2.10) can be linearized by the Bussang decomposition as follows:

$$\begin{aligned} \mathbf{y}_{\mathrm{t}} &= \mathbf{V}_{\mathrm{t}}\mathbf{P}\mathbf{h} + \mathbf{V}_{\mathrm{t}}\mathbf{z}_{\mathrm{t}} + \mathbf{d}_{\mathrm{t}} \\ &= \mathbf{A}_{\mathrm{t}}\mathbf{h} + \mathbf{n}_{\mathrm{t}} \end{aligned} \tag{2.11}$$

Table 2.1: Optimum uniform quantizer for $\mathcal{N}(0,1)$ Gaussian inputs.

Resolution b	1-bit	2-bit	3-bit	4-bit
Step size Δ_b	$\sqrt{8/\pi}$	0.996	0.586	0.335
Distortion η_b	$1-2/\pi$	0.1188	0.0374	0.0115

where $\mathbf{A}_{\text{t}} \equiv \mathbf{V}_{\text{t}}\mathbf{P}$, $\mathbf{n}_{\text{t}} \equiv \mathbf{V}_{\text{t}}\mathbf{z}_{\text{t}} + \mathbf{d}_{\text{t}}$ combines the receiver and equivalent quantization noise, and $\mathbf{V}_{\text{t}} \in \mathbb{R}^{2NT_{\text{t}}\times 2NT_{\text{t}}}$ is a diagonal matrix and given as [52]

$$\mathbf{V}_{\text{t}} = \frac{\Delta}{\sqrt{2\pi}}\operatorname{diag}(\mathbf{\Sigma}_{\mathbf{r}_{\text{t}}})^{-\frac{1}{2}} \times \sum_{i=1}^{2^b-1} \exp\Big\{-\frac{1}{2}\Delta^2(i-2^{b-1})^2\operatorname{diag}(\mathbf{\Sigma}_{\mathbf{r}_{\text{t}}})^{-1}\Big\}$$

where $\mathbf{\Sigma}_{\mathbf{r}_{\text{t}}} = \mathbf{P}\mathbf{\Sigma}_{\mathbf{h}}\mathbf{P}^T + \frac{N_0}{2}\mathbf{I}$ is the covariance matrix of $\mathbf{r}_{\text{t}} = \mathbf{P}\mathbf{h} + \mathbf{z}_{\text{t}}$. For the case of one-bit ADCs with $\Delta = \sqrt{2}$, $\mathbf{V}_{\text{t}}$ reduces to the form reported in [33, Eq. (10)].

The BMMSE channel estimator is given as [33, 52]

$$\hat{\mathbf{h}}_{\texttt{BMMSE}} = \mathbf{\Sigma}_{\mathbf{h}\mathbf{y}_{\text{t}}}\mathbf{\Sigma}_{\mathbf{y}_{\text{t}}}^{-1}\mathbf{y}_{\text{t}} = \mathbf{\Sigma}_{\mathbf{h}}\mathbf{A}_{\text{t}}^T\mathbf{\Sigma}_{\mathbf{y}_{\text{t}}}^{-1}\mathbf{y}_{\text{t}} \tag{2.12}$$

where $\mathbf{\Sigma}_{\mathbf{h}\mathbf{y}_{\text{t}}}$ is the cross-covariance matrix between $\mathbf{h}$ and $\mathbf{y}_{\text{t}}$, and $\mathbf{\Sigma}_{\mathbf{y}_{\text{t}}}$ is the covariance matrix of $\mathbf{y}_{\text{t}}$. For the case of one-bit ADCs, $\mathbf{\Sigma}_{\mathbf{y}_{\text{t}}}$ is given as [33]

$$\mathbf{\Sigma}_{\mathbf{y}_{\text{t}}} = \frac{\Delta^2}{2\pi}\arcsin\Big(\operatorname{diag}(\mathbf{\Sigma}_{\mathbf{r}_{\text{t}}})^{-\frac{1}{2}}\mathbf{\Sigma}_{\mathbf{r}_{\text{t}}}\operatorname{diag}(\mathbf{\Sigma}_{\mathbf{r}_{\text{t}}})^{-\frac{1}{2}}\Big). \tag{2.13}$$

For the case of two-bit or higher resolution ADCs, $\mathbf{\Sigma}_{\mathbf{y}_{\text{t}}}$ is given as [52]

$$\mathbf{\Sigma}_{\mathbf{y}_{\text{t}}} = \mathbf{V}_{\text{t}}\mathbf{\Sigma}_{\mathbf{r}_{\text{t}}}\mathbf{V}_{\text{t}}^T + \mathbf{\Sigma}_{\mathbf{d}_{\text{t}}}, \tag{2.14}$$

where $\mathbf{\Sigma}_{\mathbf{d}_{\text{t}}} \in \mathbb{R}^{2NT_{\text{t}}\times 2NT_{\text{t}}}$ is the covariance matrix of $\mathbf{d}_{\text{t}}$ and can be approximated as $\mathbf{\Sigma}_{\mathbf{d}_{\text{t}}} \approx \eta_b\operatorname{diag}(\mathbf{\Sigma}_{\mathbf{r}_{\text{t}}})$. The distortion factor η_b depending on the number of quantization bits b is given in Table 2.1 [76].

The BWZF channel estimator was proposed in [52] as follows:

$$\hat{\mathbf{h}}_{\texttt{BWZF}} = \left(\mathbf{A}_{\mathrm{t}}^T \operatorname{diag}(\boldsymbol{\omega}_{\mathrm{t}})\mathbf{A}_{\mathrm{t}}\right)^{-1}\mathbf{A}_{\mathrm{t}}^T \operatorname{diag}(\boldsymbol{\omega}_{\mathrm{t}})\mathbf{y}_{\mathrm{t}} \tag{2.15}$$

where $\operatorname{diag}(\boldsymbol{\omega}_{\mathrm{t}})$ is a diagonal matrix with $\boldsymbol{\omega}_{\mathrm{t}} = [\omega_{\mathrm{t},1}, \omega_{\mathrm{t},2}, \dots, \omega_{\mathrm{t},2N_{\mathrm{rx}}T_{\mathrm{t}}}]$ on the diagonal, and

$$\omega_{\mathrm{t},i} = \frac{1}{\mathbb{E}[z_{\mathrm{t},i}^2] + \mathbb{E}[d_{\mathrm{t},i}^2|y_{\mathrm{t},i}]},\ i = 1, \dots, 2N_{\mathrm{rx}}T_{\mathrm{t}}.$$

Here, $y_{\mathrm{t},i}$, $z_{\mathrm{t},i}$, and $d_{\mathrm{t},i}$ are the i-th elements of $\mathbf{y}_{\mathrm{t}}$, $\mathbf{z}_{\mathrm{t}}$, and $\mathbf{d}_{\mathrm{t}}$, respectively. The key idea of BWZF is that given an observed quantized signal vector $\mathbf{y}_{\mathrm{t}}$, the elements of $\mathbf{r}_{\mathrm{t}}$ have different variances. Exploiting this fact, the BWZF estimator sets the signals with lower variances to have higher weights.

2.3.3 Bussgang Decomposition-based Linear Data Detectors

In this section, we present several Bussgang decomposition-based linear detectors for low-resolution massive MIMO systems [52,64]. Since data detection by linear detectors does not depend on the time slot index, we consider (2.7) in a single time slot and convert it into the real domain as follows:

$$\mathbf{y}_{\mathrm{d}} = \mathcal{Q}_b\left(\mathbf{H}_{\mathrm{d}}\mathbf{x}_{\mathrm{d}} + \mathbf{z}_{\mathrm{d}}\right), \tag{2.16}$$

where

$$\mathbf{y}_{\mathrm{d}} = \begin{bmatrix} \Re\{\mathbf{y}_{\mathrm{d}}^{\mathbb{C}}\} \\ \Im\{\mathbf{y}_{\mathrm{d}}^{\mathbb{C}}\} \end{bmatrix},\ \mathbf{x}_{\mathrm{d}} = \begin{bmatrix} \Re\{\mathbf{x}_{\mathrm{d}}^{\mathbb{C}}\} \\ \Im\{\mathbf{x}_{\mathrm{d}}^{\mathbb{C}}\} \end{bmatrix},\ \mathbf{z}_{\mathrm{d}} = \begin{bmatrix} \Re\{\mathbf{z}_{\mathrm{d}}^{\mathbb{C}}\} \\ \Im\{\mathbf{z}_{\mathrm{d}}^{\mathbb{C}}\} \end{bmatrix},\ \text{and } \mathbf{H}_{\mathrm{d}} = \begin{bmatrix} \Re\{\mathbf{H}^{\mathbb{C}}\} & -\Im\{\mathbf{H}^{\mathbb{C}}\} \\ \Im\{\mathbf{H}^{\mathbb{C}}\} & \Re\{\mathbf{H}^{\mathbb{C}}\} \end{bmatrix}.$$

Note that $\mathbf{y} \in \mathbb{R}^{2N_{\mathrm{rx}}}$, $\mathbf{x} \in \mathbb{R}^{2N_{\mathrm{tx}}}$, $\mathbf{z} \in \mathbb{R}^{2N_{\mathrm{rx}}}$, and $\mathbf{H} \in \mathbb{R}^{2N_{\mathrm{rx}}\times 2N_{\mathrm{tx}}}$.

Applying the Bussgang decomposition to (2.16), we obtain

$$\begin{aligned}\mathbf{y}_{\mathrm{d}} &= \mathbf{V}_{\mathrm{d}}\mathbf{H}_{\mathrm{d}}\mathbf{x}_{\mathrm{d}} + \mathbf{V}_{\mathrm{d}}\mathbf{z}_{\mathrm{d}} + \mathbf{d}_{\mathrm{d}} \\ &= \mathbf{A}_{\mathrm{d}}\mathbf{x}_{\mathrm{d}} + \mathbf{n}_{\mathrm{d}} \end{aligned} \tag{2.17}$$

where $\mathbf{V}_{\mathrm{d}}$ is a diagonal matrix and given as

$$\mathbf{V}_{\mathrm{d}} = \frac{\Delta}{\sqrt{2\pi}} \operatorname{diag}(\boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}})^{-\frac{1}{2}} \times \sum_{i=1}^{2^b-1} \exp\left\{ -\frac{1}{2}\Delta^2 (i - 2^{b-1})^2 \operatorname{diag}(\boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}})^{-1} \right\}$$

and $\boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}} = \mathbf{H}_{\mathrm{d}}\boldsymbol{\Sigma}_{\mathbf{x}_{\mathrm{d}}}\mathbf{H}_{\mathrm{d}}^T + \frac{1}{2}N_0\mathbf{I}$. For the case of 1-bit ADCs, the covariance of $\mathbf{n}_{\mathrm{d}}$ is given in closed form as [20]

$$\begin{aligned}\boldsymbol{\Sigma}_{\mathbf{n}_{\mathrm{d}}} =& \frac{\Delta^2}{2\pi}\Big[\arcsin\Big(\operatorname{diag}(\boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}})^{-\frac{1}{2}} \boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}} \operatorname{diag}(\boldsymbol{\Sigma}_{\mathbf{r}})^{-\frac{1}{2}} \Big) - \\ & \operatorname{diag}(\boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}})^{-\frac{1}{2}} \boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}} \operatorname{diag}(\boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}})^{-\frac{1}{2}} + \frac{N_0}{2} \operatorname{diag}(\boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}})^{-1} \Big]. \end{aligned} \tag{2.18}$$

For few-bit ADCs, the covariance of $\mathbf{n}_{\mathrm{d}}$ can be approximated as $\boldsymbol{\Sigma}_{\mathbf{n}_{\mathrm{d}}} \approx \frac{N_0}{2}\mathbf{V}_{\mathrm{d}}\mathbf{V}_{\mathrm{d}}^T + \eta_b \operatorname{diag}(\boldsymbol{\Sigma}_{\mathbf{r}_{\mathrm{d}}})$. The effective noise $\mathbf{n}_{\mathrm{d}}$ is often modeled as $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{n}_{\mathrm{d}}})$. Based on this linearized model, different linear detectors such as BZF and BMMSE were introduced in [64] as follows:

$$\mathbf{W}_{\mathrm{d},\mathtt{BZF}} = \left({\mathbf{A}_{\mathrm{d}}}^H \mathbf{A}_{\mathrm{d}} \right)^{-1} {\mathbf{A}_{\mathrm{d}}}^H \tag{2.19}$$

$$\mathbf{W}_{\mathrm{d},\mathtt{BMMSE}} = {\mathbf{A}_{\mathrm{d}}}^H \left(\mathbf{A}_{\mathrm{d}}\mathbf{A}_{\mathrm{d}}^H + \boldsymbol{\Sigma}_{\mathbf{n}_{\mathrm{d}}} \right)^{-1}. \tag{2.20}$$

The authors in [52] also proposed a BWZF detector

$$\mathbf{W}_{\mathrm{d},\mathtt{BWZF}} = \left(\mathbf{A}_{\mathrm{d}}^T \operatorname{diag}(\boldsymbol{\omega}_{\mathrm{d}}) \mathbf{A}_{\mathrm{d}} \right)^{-1} \mathbf{A}_{\mathrm{d}}^T \operatorname{diag}(\boldsymbol{\omega}_{\mathrm{d}}) \tag{2.21}$$

where $\text{diag}(\boldsymbol{\omega}_{\text{d}})$ is a diagonal matrix with $\boldsymbol{\omega}_{\text{d}} = [\omega_{\text{d},1}, \omega_{\text{d},2}, \ldots, \omega_{\text{d},2N_{\text{rx}}}]$ on the diagonal, and

$$\omega_{\text{d},i} = \frac{1}{\mathbb{E}[z_{\text{d},i}^2] + \mathbb{E}[d_{\text{d},i}^2 | y_{\text{d},i}]}, \; i = 1, \ldots, 2N_{\text{rx}}.$$

2.4 Concluding Remarks

There have been numerous existing methods for channel estimation and data detection in MIMO systems with low-resolution ADCs. However, these methods often suffer from drawbacks such as high-computational complexity, system scalability, non-robustness, or limited performance. For example, the ML, OSD, and GAMP-based methods have too high computational complexities for practical implementation. Both ML and nML methods are non-robust when the CSI is not known perfectly. Several blind detection methods do not require CSI but have the scalability issue. The Bussgang decomposition-based linear receivers presented in the previous section are less computationally complicated, more robust and scalable but they have limited performance.

The work in this dissertation exploits machine learning to address the above issues. This is motivated by the fact that machine learning has been shown in practice to be a very powerful tool for solving non-linear problems. Since the low-resolution ADCs are severely non-linear, it is of significant importance and interest to take advantage of machine learning for low-resolution MIMO signal processing problems.

Chapter 3

Supervised and semi-supervised learning for MIMO blind detection with low-resolution ADCs

3.1 Introduction

This chapter focuses on the blind detection problem in MIMO systems with low-resolution ADCs. Blind detection in this context means detection without information about the CSI. The authors of [65, 79] proposed three supervised learning methods, referred to as empirical-Maximum-Likelihood Detection (eMLD), Minimum-Mean-Distance Detection (MMD), and Minimum-Center-Distance Detection (MCD). These blind detection methods are simple and

The materials presented in Chapter 3 have been presented at the 2018 IEEE International Conference on Communications (ICC) in Kansas City, MO, USA [77] and published in the IEEE Transactions on Wireless Communications [78].

easy to implement, but their efficiency is heavily dependent on the training sequence. When the length of the training sequence is short, the learned results do not correctly describe the input-output relations of the system. Based on this observation, we propose in this chapter two efficient learning methods to resolve the problem of short training sequences. Since MCD outperforms eMLD and MMD, and the complexity of MCD is also lower than that of eMLD and MMD, we compare our proposed methods to MCD only. In this chapter, we provide a complete analysis of the proposed methods and make the following contributions:

- We propose two learning methods that are capable of achieving more precise input-output relations compared to [65, 79] given the same training sequence, and hence will improve the detection accuracy. The first method exploits the use of the CRC to acquire more training data. In the second method, no CRC is required, but the to-be-decoded data is self-classified into groups, which help improve the learned results. This method is based on the K-means clustering technique. However, unlike the detection method in [80], which is specifically designed to work with Space Shift Keying modulation and only one transmit antenna is active in each time slot; our method is applicable for more common modulation schemes, such as BPSK or QPSK, and all transmit antennas are active in each time slot which enables spatial multiplexing gains. In addition, the proposed method takes into account the symmetrical structure of the transmitted signal space to help improve the learned results.
- The proposed learning methods are applicable for detection with 1-bit or few-bit ADCs. We show via simulations that the proposed methods are more robust than MCD in terms of the training sequence length. Particularly, for extremely short training sequences, the performance of MCD is degraded significantly while that of our proposed methods is more stable. For example, in a system with 2 transmit antennas, 16 receive antennas, and BPSK modulation, the gain in bit error rate (BER) produced by the proposed methods can be up to 7-8 dB for BERs between 10^{-3} and 10^{-5}. Even for mod-

erately long training sequences, the gain of our proposed methods is still considerable, between 3-dB and 4-dB.

- We provide performance analyses of the VER for the case of 1-bit ADCs at both low and high signal-to-noise ratios (SNRs). Assuming perfectly learned input-output relations, we first approximate the pairwise VER at low SNR by using the Bussgang decomposition and use this approximation to derive an upper bound on the VER. The asymptotic VER performance at infinite SNR for Rayleigh fading channels is then analyzed. Simulation results confirm the accuracy of our analyses at both low and high SNRs.

- Finally, based on the performance analysis, we propose a criterion for designing transmitted signals when only a portion of all possible signals are used for transmission.

The rest of this chapter is organized as follows. The system model is first presented in Section 3.2 and the blind detection problem is stated in Section 3.3. Then, a supervised learning method and a semi-supervised learning method are proposed in Section 3.4. A performance analysis for the case of 1-bit ADCs and a criterion for transmit signal design are presented in Section 3.5. Simulations and results can be found in Section 3.6. We conclude the chapter in Section 3.7.

3.2 System Model

The considered MIMO system, as illustrated in Figure 3.1, has N_{tx} transmit antennas and N_{rx} receive antennas, where it is assumed that $N_{\text{rx}} \geq N_{\text{tx}}$. Let $\mathbf{x}^{\mathbb{C}}[n] = [x_1^{\mathbb{C}}[n], \ldots, x_{N_{\text{tx}}}^{\mathbb{C}}[n]]^T \in \mathbb{C}^{N_{\text{tx}}}$ be the transmitted signal vector at time slot n, where $x_i^{\mathbb{C}}[n]$ is the symbol transmitted at the i^{th} transmit antenna. Each symbol $x_i^{\mathbb{C}}[n]$ is drawn from a constellation $\mathcal{M}^{\mathbb{C}}$ with a constellation size of $M = |\mathcal{M}^{\mathbb{C}}|$ under the power constraint $\mathbb{E}[|x_i^{\mathbb{C}}[n]|^2] = 1$. The channel is

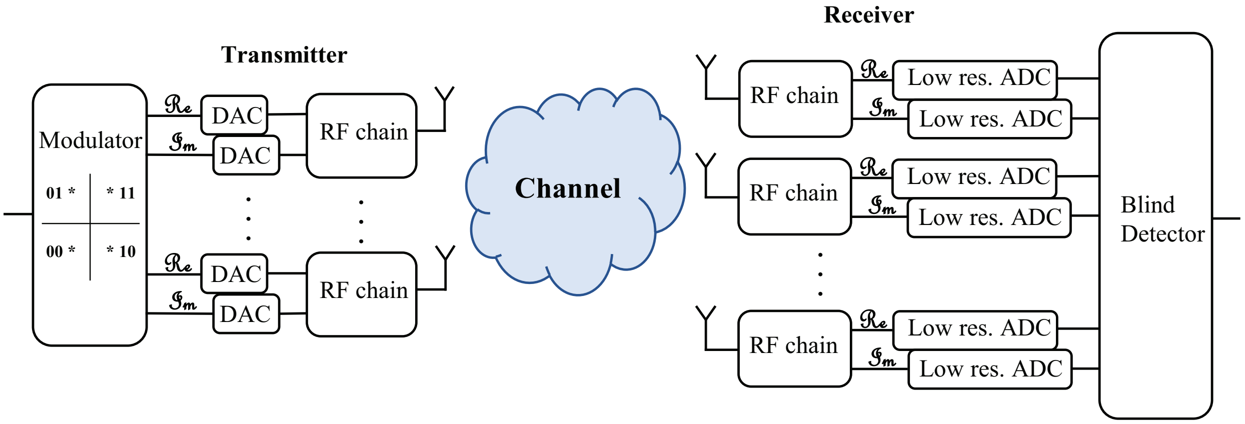


Figure 3.1: Block diagram of a MIMO communication system with low-resolution ADC at the receiver.

assumed to be block-fading, and each block-fading interval lasts for T_b time slots. Hence, the channel $\mathbf{H}^\mathbb{C} \in \mathbb{C}^{N_\mathrm{rx} \times N_\mathrm{tx}}$ remains constant over T_b time slots. For the analysis and simulations, we assume that the elements of $\mathbf{H}^\mathbb{C}$ are independent and identically distributed (i.i.d.) as $\mathcal{CN}(0,1)$, but the proposed algorithms are applicable to any channel model. The system model in each block-fading interval is

$$\mathbf{r}^\mathbb{C}[n] = \mathbf{H}^\mathbb{C}\mathbf{x}^\mathbb{C}[n] + \mathbf{z}^\mathbb{C}[n], \tag{3.1}$$

where $\mathbf{r}^\mathbb{C}[n] = [r_1^\mathbb{C}[n], \ldots, r_{N_\mathrm{rx}}^\mathbb{C}[n]]^T \in \mathbb{C}^{N_\mathrm{rx}}$ is the analog received signal vector, and $\mathbf{z}^\mathbb{C}[n] = [z_1^\mathbb{C}[n], \ldots, z_{N_\mathrm{rx}}^\mathbb{C}[n]]^T \in \mathbb{C}^{N_\mathrm{rx}}$ is the noise vector. The noise elements are assumed to be i.i.d. with $z_i^\mathbb{C}[n] \sim \mathcal{CN}(0, N_0)$. CSI is unavailable at both the transmitter and receiver sides, i.e., $\mathbf{H}^\mathbb{C}$ is unknown. The SNR is defined as $\varrho = N_\mathrm{tx}/N_0$.

The real and imaginary parts of each received symbol are applied to two separate ADCs. Hence, if $\mathbf{y}^\mathbb{C}[n] = \left[y_1^\mathbb{C}[n], \ldots, y_{N_\mathrm{rx}}^\mathbb{C}[n]\right]^T \in \mathbb{C}^{N_\mathrm{rx}}$ is the quantized version of the received signal vector $\mathbf{r}^\mathbb{C}[n]$, then $\mathbf{y}^\mathbb{C}[n] = Q_b(\mathbf{r}^\mathbb{C}[n])$ in which $\Re\{y_i^\mathbb{C}[n]\} = Q_b(\Re\{r_i^\mathbb{C}[n]\})$ and $\Im\{y_i^\mathbb{C}[n]\} = Q_b(\Im\{r_i^\mathbb{C}[n]\})$ for all $i \in \mathcal{N}_\mathrm{rx} = \{1, 2, \ldots, N_\mathrm{rx}\}$. The optimal step size for the quantizer $\mathcal{Q}_b(\cdot)$ in the considered system is $\Delta_\mathrm{opt} = \sqrt{(N_\mathrm{tx} + N_0)/2}\Delta_\mathrm{opt}^\mathrm{standard}$. The variance of the analog received signals $N_\mathrm{t} + N_0$ is assumed to be known at the receiver. It should be noted that this mid-rise uniform quantizer satisfies $Q_b(-r) = -Q_b(r), \forall r$.

Since the derivations this chapter are mainly in the complex domain, for notational simplicity, we drop the superscript $^{\mathbb{C}}$ in all notations in the rest of this chapter.

3.3 Blind Detection Problem

This section describes the blind detection problem for the block-fading channel. The first T_{t} time slots of each block fading interval contain the training symbol sequence while the remaining $T_{\mathrm{d}} = T_{\mathrm{b}} - T_{\mathrm{t}}$ time slots comprise the data symbol sequence. Let $\check{\mathcal{X}} = \{\check{\mathbf{x}}_1, \check{\mathbf{x}}_2, \dots, \check{\mathbf{x}}_K\}$ denote the set of all possible transmitted symbol vectors with $K = M^{N_{\mathrm{tx}}}$ and let $\mathcal{K} = \{1, 2, \dots, K\}$. Hereafter, a possible transmitted symbol vector is called a *label*. We first revisit the MCD method presented in [79], which serves as a baseline for the study of this chapter. The input-output relations to be learned in the MCD method are $\{\mathbb{E}[\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_k], k \in \mathcal{K}\}$, in which $\mathbb{E}[\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_k]$ represents the centroid of the received quantized signal given that the label $\check{\mathbf{x}}_k$ is transmitted. The MCD data detection is given by

$$f(\mathbf{y}[n]) = \underset{k \in \mathcal{K}}{\operatorname{argmin}} \left\| \mathbf{y}[n] - \mathbb{E}[\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_k] \right\|_2, \tag{3.2}$$

where $\mathbf{y}[n]$ is the received data symbol vector at time slot n with $n \in \{T_{\mathrm{t}}+1, \dots, T_{\mathrm{b}}\}$. Thus, the MCD approach identifies the index of the transmitted label as the one whose centroid is closest to the received vector. Denote $\check{\mathbf{y}}_k = \mathbb{E}[\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_k]$; each $\check{\mathbf{y}}_k$ is called a *representative vector* for the label $\check{\mathbf{x}}_k$. There are K representative vectors $\check{\mathcal{Y}} = \{\check{\mathbf{y}}_1, \check{\mathbf{y}}_2, \dots, \check{\mathbf{y}}_K\}$. Thus, the MCD method has to learn $\check{\mathcal{Y}}$ in order to perform the detection task. We now present two MCD training methods from [65, 79] that help the receiver empirically learn $\check{\mathcal{Y}}$.

3.3.1 Full-space Training Method

Since the transmitted signal space $\check{\mathcal{X}}$ contains K labels, a straightforward method to help the receiver learn $\check{\mathcal{Y}}$ is using a training sequence that contains all the labels, where each label is repeated a number of times. Hence, the training symbol matrix can be represented as $\mathbf{X}_{\rm t} = [\check{\mathbf{X}}_1, \check{\mathbf{X}}_2, \ldots, \check{\mathbf{X}}_K]$, where $\check{\mathbf{X}}_k = [\check{\mathbf{x}}_k, \ldots, \check{\mathbf{x}}_k] \in \mathbb{C}^{N_{\rm tx} \times L_{\rm t}}$ consists of $L_{\rm t}$ labels $\check{\mathbf{x}}_k$, $k \in \mathcal{K}$. Using this training method, the representative vector $\check{\mathbf{y}}_k$ can be learned empirically as

$$\check{\mathbf{y}}_k = \frac{1}{L_{\rm t}} \sum_{t=1}^{L_{\rm t}} \mathbf{y}[(k-1)L_{\rm t} + t], \tag{3.3}$$

where $\mathbf{Y}_{\rm t} = \big[\mathbf{y}[1], \ldots, \mathbf{y}[T_{\rm t}]\big] = Q_b(\mathbf{H}\mathbf{X}_{\rm t} + \mathbf{Z}_{\rm t})$. The length of the training sequence is $T_{\rm t} = KL_{\rm t}$. This training method has been employed in [79].

3.3.2 Subspace Training Method

It is worth noting that the training sequence does not need to cover all the labels for the receiver to learn $\check{\mathcal{Y}}$ when $\mathcal{M}$ satisfies either of the following two conditions:

- *Condition* 1: $-x \in \mathcal{M}$, $\forall x \in \mathcal{M}$.
- *Condition* 2: $\alpha x \in \mathcal{M}$, $\forall x \in \mathcal{M}$ and $\forall \alpha \in \{-1, j, -j\}$.

Although Condition 2 implies Condition 1 when $\alpha = -1$, i.e., any $\mathcal{M}$ satisfying Condition 2 will also satisfy Condition 1, we maintain these as two separate conditions for convenience in our later derivations. Examples of $\mathcal{M}$ for Condition 1 are BPSK, 8-QAM and for Condition 2 are QPSK, 16-QAM.

If Condition 1 is satisfied, $-\check{\mathbf{x}}_k \in \check{\mathcal{X}}$ for all $\check{\mathbf{x}}_k \in \check{\mathcal{X}}$. The set of all labels can be written as

$$\check{\mathcal{X}} = \{\check{\mathcal{X}}_{\text{ha}}, -\check{\mathcal{X}}_{\text{ha}}\}, \tag{3.4}$$

where $\check{\mathcal{X}}_{\text{ha}} = \{\check{\mathbf{x}}_1, \ldots, \check{\mathbf{x}}_{K/2}\}$. Without loss of generality, it is assumed that $\check{\mathbf{x}}_{k+K/2} = -\check{\mathbf{x}}_k$ with $k \in \{1, \ldots, K/2\}$. If Condition 2 is satisfied, then $\alpha\check{\mathbf{x}}_k \in \check{\mathcal{X}}$ for all $\check{\mathbf{x}}_k \in \check{\mathcal{X}}$ and $\alpha \in \{-1, j, -j\}$. The set of all labels can be written as

$$\check{\mathcal{X}} = \{\check{\mathcal{X}}_{\text{fo}}, -\check{\mathcal{X}}_{\text{fo}}, j\check{\mathcal{X}}_{\text{fo}}, -j\check{\mathcal{X}}_{\text{fo}}\}, \tag{3.5}$$

where $\check{\mathcal{X}}_{\text{fo}} = \{\check{\mathbf{x}}_1, \ldots, \check{\mathbf{x}}_{K/4}\}$. It is then assumed that $\check{\mathbf{x}}_{k+K/4} = -\check{\mathbf{x}}_k$, $\check{\mathbf{x}}_{k+K/2} = j\check{\mathbf{x}}_k$, and $\check{\mathbf{x}}_{k+3K/4} = -j\check{\mathbf{x}}_k$ for $k \in \{1, \ldots, K/4\}$. The subscripts 'ha' and 'fo' here stand for 'half' and 'fourth', indicating the first one-half and the first one-fourth of the set $\check{\mathcal{X}}$, respectively.

The work in [65] showed that if the transmitter employs QAM modulation and the quantization function satisfies $Q_b(-r) = -Q_b(r)$ for any $r \in \mathbb{R}$, then the length of the training sequence can be reduced to $T_{\text{t}} = KL_{\text{t}}/4$. In Proposition 3.1 below, we generalize this result for any modulation scheme.

Proposition 3.1. *Given any constellation $\mathcal{M}$, if the quantizer $Q_b(.)$ is symmetric, i.e., $Q_b(-r) = -Q_b(r)\ \forall r \in \mathbb{R}$, the length of the training sequence T_{t} can be reduced to*

$$T_{\text{t}} = \begin{cases} \frac{1}{2}KL_{\text{t}} & \text{if Condition 1 holds,} \\ \frac{1}{4}KL_{\text{t}} & \text{if Condition 2 holds.} \end{cases} \tag{3.6}$$

Proof. For any two labels $\check{\mathbf{x}}_{k_1}$ and $\check{\mathbf{x}}_{k_2} = -\check{\mathbf{x}}_{k_1}$, we have

$$p(\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_{k_2}) = \mathbb{P}\big[\mathbf{y} = Q_b(\mathbf{H}\mathbf{x}_{k_2} + \mathbf{z})\big] = \mathbb{P}\big[\mathbf{y} = Q_b(-\mathbf{H}\mathbf{x}_{k_1} - \mathbf{z})\big] = \mathbb{P}\big[-\mathbf{y} = Q_b(\mathbf{H}\mathbf{x}_{k_1} + \mathbf{z})\big]$$
$$= p(-\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_{k_1}). \tag{3.7}$$

Therefore, $\check{\mathbf{y}}_{k_2} = -\check{\mathbf{y}}_{k_1}$ since

$$\begin{aligned}\check{\mathbf{y}}_{k_2} = \mathbb{E}\big[\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_{k_2}\big] &= \sum \mathbf{y} p(\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_{k_2}) = \sum \mathbf{y} p(-\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_{k_1}) \\ &= -\sum \dot{\mathbf{y}} p(\dot{\mathbf{y}}|\mathbf{x} = \check{\mathbf{x}}_{k_1}) \qquad (3.8) \\ &= -\mathbb{E}\big[\mathbf{y}|\mathbf{x} = \check{\mathbf{x}}_{k_1}\big] = -\check{\mathbf{y}}_{k_1}, \qquad (3.9)\end{aligned}$$

where (3.8) is obtained by setting $\dot{\mathbf{y}} = -\mathbf{y}$ and (3.9) holds because the sample spaces of $\dot{\mathbf{y}}$ and $\mathbf{y}$ are the same. Hence, the representative vectors satisfy $\check{\mathbf{y}}_{k+K/2} = -\check{\mathbf{y}}_k$ with $k \in \{1, \ldots, K/2\}$ if Condition 1 holds. This means the training sequence only needs to cover $\check{\mathcal{X}}_{\text{ha}}$ to help the receiver learn all K representative vectors in $\check{\mathcal{Y}}$. Similarly, when Condition 2 holds, we can also show that $\check{\mathbf{y}}_{k+K/4} = -\check{\mathbf{y}}_k$, $\check{\mathbf{y}}_{k+K/2} = j\check{\mathbf{y}}_k$, and $\check{\mathbf{y}}_{k+3K/4} = -j\check{\mathbf{y}}_k$ with $k \in \{1, \ldots, K/4\}$, and so the training sequence only needs to contain $\check{\mathcal{X}}_{\text{fo}}$. It should be noted that the proof for Condition 2 requires that $Q_b(jc) = jQ_b(c), \forall c \in \mathbb{C}$, which is satisfied by the quantizer being used. □

3.4 Proposed Learning Methods

The MCD detection method is simple but it has a primary drawback – its detection accuracy heavily depends on the length of the training sequence. If the training sequence cannot provide accurate representative vectors in (3.3), then detection errors will appear in (3.2). In fact, a short training sequence often results in poor estimation of the representative vectors. In order to improve the detection accuracy *without* lengthening the training sequence, the

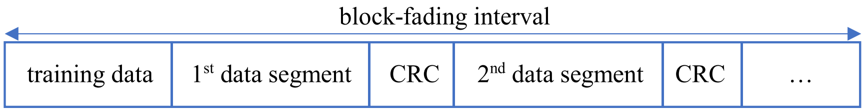


Figure 3.2: Usage of CRC for multiple data segments in each block-fading interval.

idea is to use the training sequence as an initial guide for the learning process, and then find more precise representative vectors by exploiting other information.

3.4.1 Proposed Supervised Learning Method

In practical communications systems, error control mechanisms such as the CRC can be used to determine whether a segment of data is correctly decoded or not. This approach has been exploited to mitigate the effect of imperfect CSI on the ML detection for low-resolution ADCs [81, 82]. An error correcting code was also used to update the weights in a neural network as the channel changes, assuming perfect ADCs [83].

In the proposed method, should the CRC be available, it can be exploited for blind detection as follows: Data detection is first performed by the MCD using the training sequence, then the correctly decoded data confirmed by the CRC is used to augment the training set. As a result, the representative vectors obtained from the training sequence in (3.3) can be refined and the incorrectly decoded data can be re-evaluated by the MCD data detection. The process of CRC checking, updating the representative vectors, and data detection is repeated until no further correctly decoded segment is found.

In the system considered, we assume the use of the CRC for multiple data segments as illustrated in Figure 3.2. Suppose there are S segments in one block-fading interval, and each segment contains a data segment and a CRC block. Let L_{CRC} and L_{data} denote the length of the CRC and the length of each data segment in bits, respectively. Thus, we have

$$S \times (L_{\mathrm{data}} + L_{\mathrm{CRC}}) = T_{\mathrm{d}} \times N_{\mathrm{tx}} \times \log_2(M). \tag{3.10}$$

Algorithm 1: Supervised Learning Decoding.

```
1  Set $u_n = \lfloor (n-1)/L_\text{t} \rfloor + 1$ and $c_n = 1$ for $1 \leq n \leq T_\text{t}$;
2  Initialize $u_n = 0$ and $c_n = 0$ for $T_\text{t} < n \leq T_\text{b}$;
3  Set $\mathcal{C} = \varnothing$, $\mathcal{S} = \{1, 2, \ldots, S\}$, $iter = 0$, and $done = false$;
4  Find $\check{\mathcal{Y}}$ using (3.11) with the above inital setting;
5  while $done = false$ do
6      foreach $s \in \mathcal{S}$ do
7          foreach $\mathbf{y}[n] \in \mathbf{Y}_s$ do
8              Set $u_n = f(\mathbf{y}[n])$;
9          end
10         if CRC confirms the correct detection of $\mathbf{Y}_s$ then
11             Set $\mathcal{C} = \mathcal{C} \cup \{s\}$;
12             foreach $\mathbf{y}[n] \in \mathbf{Y}_s$ do
13                 Set $c_n = 1$;
14             end
15         end
16         Update $\check{\mathcal{Y}}$ using (3.11);
17     end
18     Set $iter = iter + 1$;
19     Set $\mathcal{S} = \mathcal{S} \backslash \mathcal{C}$, then set $\mathcal{C} = \varnothing$;
20     if $\mathcal{S} = \varnothing$ or $iter = iter_\text{max}$ or no change in $\mathbf{u}$ then
21         $done = true$;
22     end
23 end
```

We also assume that $L_\text{data} + L_\text{CRC}$ is a multiple of $N_\text{tx}\text{log}_2 M$. This means the number of bits in a segment is a multiple of the number bits in a transmitted vector. The decoding algorithm of this proposed method is presented in Algorithm 1. The detailed explanation of Algorithm 1 is as follows.

Let $\mathbf{u} = [u_1, \ldots, u_{T_\text{b}}]$ denote the vector of decoded indices where $u_n \in \mathcal{K}$ with $n \in \{1, \ldots, T_\text{b}\}$ is the decoded index of received signal $\mathbf{y}[n]$. Here, we can set $u_n = \lfloor (n-1)/L_\text{t} \rfloor + 1$ for $1 \leq n \leq T_\text{t}$ (line 1) due to the training sequence and we can initialize $u_n = 0$ for $T_\text{t} < n \leq T_\text{b}$ (line 2). Let $\mathbf{c} = [c_1, \ldots, c_{T_\text{b}}]$ denote the vector of binary values where $c_n = 1$ if the CRC confirms a correct detection of $\mathbf{y}[n]$, otherwise $c_n = 0$. Note that $c_n = 0$ does not imply an incorrect detection of $\mathbf{y}[n]$. Instead, it implies that the CRC cannot confirm a correct detection of $\mathbf{y}[n]$. Since the first T_t time slots are for the training sequence, we can set $c_n = 1$

for $1 \leq n \leq T_{\mathrm{t}}$ (line 1) and initialize $c_n = 0$ for $T_{\mathrm{t}} < n \leq T_{\mathrm{b}}$ (line 2). Let s denote the index of the segments, $s \in \{1, \ldots, S\}$, and let $\mathbf{Y}_s$ denote the s^{th} received data segment. After the detection of each segment, $\check{\mathbf{y}}_k$ can be refined as (line 16):

$$\check{\mathbf{y}}_k = \frac{\sum_{n=1}^{T_{\mathrm{b}}} \big(\mathbb{I}[u_n = k] + c_n \gamma(n,k)\big)\mathbf{y}[n]}{\sum_{n=1}^{T_{\mathrm{b}}} \big(\mathbb{I}[u_n = k] + c_n \mathbb{I}[\gamma(n,k) \neq 0]\big)} \tag{3.11}$$

where $\mathbb{I}$ is the indicator function, and $\gamma(n,k)$ is a function of n and k defined as follows:

- *Condition* 1: $\gamma(n,k) = -\mathbb{I}[u_n = \bar{k}]$ with

$$\bar{k} = \begin{cases} k + \frac{K}{2} & \text{if } k \leq \frac{K}{2}, \\ k - \frac{K}{2} & \text{if } k > \frac{K}{2}. \end{cases} \tag{3.12}$$

- *Condition* 2:

  Let $\mathcal{K}_1 = \left\{1, \ldots, \frac{K}{4}\right\}$, $\mathcal{K}_2 = \left\{\frac{K}{4} + 1, \ldots, \frac{K}{2}\right\}$, $\mathcal{K}_3 = \left\{\frac{K}{2} + 1, \ldots, \frac{3K}{4}\right\}$, and $\mathcal{K}_4 = \left\{\frac{3K}{4} + 1, \ldots, K\right\}$;

$$\begin{aligned} &\text{if } k \in \mathcal{K}_1, \text{ let } \bar{k}_1 = k + \frac{K}{4}, \bar{k}_2 = k + \frac{K}{2}, \bar{k}_3 = k + \frac{3K}{4}, \\ &\text{if } k \in \mathcal{K}_2, \text{ let } \bar{k}_1 = k - \frac{K}{4}, \bar{k}_2 = k + \frac{K}{2}, \bar{k}_3 = k + \frac{K}{4}, \\ &\text{if } k \in \mathcal{K}_3, \text{ let } \bar{k}_1 = k + \frac{K}{4}, \bar{k}_2 = k - \frac{K}{4}, \bar{k}_3 = k - \frac{K}{2}, \\ &\text{if } k \in \mathcal{K}_4, \text{ let } \bar{k}_1 = k - \frac{K}{4}, \bar{k}_2 = k - \frac{3K}{4}, \bar{k}_3 = k - \frac{K}{2}, \end{aligned}$$

$$\gamma(n,k) = -\mathbb{I}[u_n = \bar{k}_1] - j\mathbb{I}[u_n = \bar{k}_2] + j\mathbb{I}[u_n = \bar{k}_3]. \tag{3.13}$$

Intuitively, the representative vector $\check{\mathbf{y}}_k$ in (3.11) is updated by using received vectors whose decoded indices are k and ones that are decoded correctly (confirmed by the CRC) with decoded indices $\bar{k}$ for Condition 1 or $\bar{k}_1$, $\bar{k}_2$, $\bar{k}_3$ for Condition 2.

The refined representative vectors are then used to perform data detection on the next segment (back to lines 7–9). In the first iteration, the next segment is $\mathbf{Y}_{s+1}$, which has not been decoded before. In the subsequent iterations, the next segment is one that has not been successfully decoded. Iterations here are accounted for by the **while** loop. The process of CRC checking, updating the representative vectors and data detection is repeated until all segments are decoded correctly or no change in $\mathbf{u}$ is found or a maximum number of iterations is reached (line 20).

3.4.2 Proposed Semi-supervised Learning Method

In this part we propose a semi-supervised learning method. This proposed method is based on the K-means clustering technique [84]. The idea is to use the training sequence as an initial guidance to find coarse estimates of the representative vectors. Based on these coarse estimates, the received data vectors are then self-classified iteratively.

The K-means clustering technique aims to partition data into a number of clusters. However, in this communication context, the decoding task is not just to partition the received data into clusters but also to assign labels to the clusters, which can be done by using the training sequence. In addition, we take into account the constraints $\check{\mathbf{y}}_{k+K/2} = -\check{\mathbf{y}}_k$, $k = 1, \ldots, K/2$, if Condition 1 holds; and the constraints $\check{\mathbf{y}}_{k+K/4} = -\check{\mathbf{y}}_k$, $\check{\mathbf{y}}_{k+K/2} = j\check{\mathbf{y}}_k$, $\check{\mathbf{y}}_{k+3K/4} = -j\check{\mathbf{y}}_k$, $k = 1, \ldots, K/4$, if Condition 2 holds. These constraints can be adopted because clusters are formed based on their centroids, which are also referred to as the representative vectors $\{\check{\mathbf{y}}_k\}$ in this work.

First, we introduce a set of binary variables $\beta_{n,k} \in \{0, 1\}$ to indicate which of the K labels that the received vector $\mathbf{y}[n]$ belongs to. Specifically, if a received vector $\mathbf{y}[n]$ belongs to label k, then $\beta_{n,k} = 1$ and $\beta_{n,l} = 0$ $\forall l \neq k$. We have the following optimization problems:

- *Condition 1*:

$$\begin{aligned}&\underset{\{\beta_{n,k}\},\{\check{\mathbf{y}}_k\}}{\text{minimize}} \quad J = \sum_{n=1}^{T_\mathrm{b}} \sum_{k=1}^{K} \beta_{n,k} \|\mathbf{y}[n] - \check{\mathbf{y}}_k\|^2 \\ &\text{subject to} \quad \check{\mathbf{y}}_{k+\frac{K}{2}} = -\check{\mathbf{y}}_k, \quad k = 1, \ldots, K/2.\end{aligned} \tag{3.14}$$

The objective function in (3.14) is called the *distortion measure* [84]. This problem can be rewritten as

$$\underset{\{\beta_{n,k}\},\{\check{\mathbf{y}}_k\}}{\text{minimize}} \quad J_1 \tag{3.15}$$

where

$$J_1 = \sum_{n=1}^{T_\mathrm{b}} \sum_{k=1}^{\frac{K}{2}} \left(\beta_{n,k} \|\mathbf{y}[n] - \check{\mathbf{y}}_k\|^2 + \beta_{n,k+\frac{K}{2}} \|\mathbf{y}[n] + \check{\mathbf{y}}_k\|^2 \right). \tag{3.16}$$

Problem (3.15) can be solved iteratively in which each iteration finds $\{\beta_{n,k}\}$ based on fixed $\{\check{\mathbf{y}}_k\}$ and vice versa. If $\{\check{\mathbf{y}}_k\}$ are fixed, J_1 is a linear function of $\{\beta_{n,k}\}$. It can be seen that the solutions $\{\beta_{n,k}\}$ are independent of n, so they can be found separately. With any $n \in \{T_\mathrm{t} + 1, \ldots, T_\mathrm{b}\}$, the optimization problem for $\{\beta_{n,k}\}$ is

$$\underset{\{\beta_{n,k}\}}{\text{minimize}} \quad \sum_{k=1}^{K} \beta_{n,k} \|\mathbf{y}[n] - \check{\mathbf{y}}_k\|^2, \tag{3.17}$$

whose solution is found by setting $\beta_{n,k} = 1$ for the k associated with the minimum value of $\|\mathbf{y}[n] - \check{\mathbf{y}}_k\|^2$. The solutions $\{\beta_{n,k}\}$ can be written as

$$\beta_{n,k} = \begin{cases} 1 & \text{if } k = \operatorname{argmin}_{k'} \|\mathbf{y}[n] - \check{\mathbf{y}}_{k'}\|^2, \\ 0 & \text{otherwise.} \end{cases} \tag{3.18}$$

It should be noted that $\beta_{n,k} = 1$ whenever $n \leq T_\mathrm{t}$ and $k = \lfloor (n-1)/L_\mathrm{t} \rfloor + 1$ because the labels of the received training vectors are known at the receiver. When the $\{\beta_{n,k}\}$ are fixed, J_1 becomes a quadratic function of $\{\check{\mathbf{y}}_k\}$. Hence the solutions $\{\check{\mathbf{y}}_k\}$ can be found by finding the derivative of J_1 with respect to $\check{\mathbf{y}}_k$:

$$\frac{\partial J_1}{\partial \check{\mathbf{y}}_k} = \sum_{n=1}^{T_\mathrm{b}} \beta_{n,k}\big(-\mathbf{y}[n]^H + \check{\mathbf{y}}_k^H\big) + \beta_{n,k+\frac{K}{2}}\big(\mathbf{y}[n]^H + \check{\mathbf{y}}_k^H\big), \tag{3.19}$$

when being set to 0 yields

$$\check{\mathbf{y}}_k = \frac{\sum_n \big(\beta_{n,k} - \beta_{n,k+\frac{K}{2}}\big)\mathbf{y}[n]}{\sum_n \big(\beta_{n,k} + \beta_{n,k+\frac{K}{2}}\big)}, \quad k = 1, \ldots, \frac{K}{2}. \tag{3.20}$$

Equation (3.20) says that the representative vector $\check{\mathbf{y}}_k$, with $k \leq K/2$, is calculated by using the received vectors that not only belong to cluster k but also to cluster $k+K/2$.

- *Condition 2*:

$$\begin{aligned} \underset{\{\beta_{n,k}\},\{\check{\mathbf{y}}_k\}}{\text{minimize}} \quad & J = \sum_{n=1}^{T_\mathrm{b}} \sum_{k=1}^{K} \beta_{n,k} \|\mathbf{y}[n] - \check{\mathbf{y}}_k\|^2 \\ \text{subject to} \quad & \check{\mathbf{y}}_{k+\frac{K}{4}} = -\check{\mathbf{y}}_k, \quad \check{\mathbf{y}}_{k+\frac{K}{2}} = j\check{\mathbf{y}}_k, \quad \check{\mathbf{y}}_{k+\frac{3K}{4}} = -j\check{\mathbf{y}}_k \\ & k = 1, \ldots, K/4. \end{aligned} \tag{3.21}$$

The optimization problem (3.21) can also be rewritten as

$$\underset{\{\beta_{n,k}\},\{\check{\mathbf{y}}_k\}}{\text{minimize}} \quad J_2 \tag{3.22}$$

where

$$\begin{aligned} J_2 = \sum_{n=1}^{T_\mathrm{b}} \sum_{k=1}^{\frac{K}{4}} \big(&\beta_{n,k}\|\mathbf{y}[n] - \check{\mathbf{y}}_k\|^2 + \beta_{n,k+\frac{K}{4}}\|\mathbf{y}[n] + \check{\mathbf{y}}_k\|^2 \\ &+ \beta_{n,k+\frac{K}{2}}\|\mathbf{y}[n] - j\check{\mathbf{y}}_k\|^2 + \beta_{n,k+\frac{3K}{4}}\|\mathbf{y}[n] + j\check{\mathbf{y}}_k\|^2\big) \end{aligned} \tag{3.23}$$

Applying the same technique as in Condition 1 to this problem, we can find $\beta_{n,k}$ from

Algorithm 2: Semi-supervised Learning Decoding.

```
1  Initialize $done = false$, $iter = 0$;
2  Find $\mathcal{Y}$ using the training sequence;
3  while $done = false$ do
4      $iter = iter + 1$;
5      Perform (3.18);
6      if Condition 1 holds then
7          Perform (3.20);
8          Set $\check{\mathbf{y}}_{k+\frac{K}{2}} = -\check{\mathbf{y}}_k$, with $k = 1, \ldots, K/2$;
9      end
10     if Condition 2 holds then
11         Perform (3.24);
12         Set $\check{\mathbf{y}}_{k+\frac{K}{4}} = -\check{\mathbf{y}}_k$, $\check{\mathbf{y}}_{k+\frac{K}{2}} = j\check{\mathbf{y}}_k$, $\check{\mathbf{y}}_{k+\frac{3K}{4}} = -j\check{\mathbf{y}}_k$, with $k = 1, \ldots, K/4$;
13     end
14     if convergent or $iter = iter_{\max}$ then
15         $done = true$;
16     end
17 end
```

(3.18) and

$$\check{\mathbf{y}}_k = \frac{\sum_n \left(\beta_{n,k} - \beta_{n,k+\frac{K}{4}} - j\beta_{n,k+\frac{K}{2}} + j\beta_{n,k+\frac{3K}{4}}\right)\mathbf{y}[n]}{\sum_n \left(\beta_{n,k} + \beta_{n,k+\frac{K}{4}} + \beta_{n,k+\frac{K}{2}} + \beta_{n,k+\frac{3K}{4}}\right)}, \quad k = 1, \ldots, \frac{K}{4}. \tag{3.24}$$

Equation (3.24) also points out that the representative vector $\check{\mathbf{y}}_k$, with $k \leq K/4$, is found by using the received vectors that not only belong to cluster k but also to clusters $k + K/4$, $k + K/2$ and $k + 3K/4$.

The decoding algorithm for this semi-supervised learning method is presented in Algorithm 2. Coarse estimation of the representative vectors is first obtained by using the training sequence (line 2). Then clustering is applied on all of the received data vectors (line 5). Depending on whether Condition 1 or Condition 2 is satisfied, the representative vectors are updated (lines 7-8 or lines 11-12). The process of clustering the received data vectors and updating the representative vectors is repeated until convergence or the number of iterations exceeds a maximum value (line 15). Convergence is achieved if the solutions $\{\beta_{n,k}\}$ are the same for two successive iterations. Convergence of Algorithm 2 is assured because after each iteration, the value of the objective function does not increase. However, the point of convergence is not guaranteed to be a global optimum.

3.5 Performance Analysis with One-bit ADCs

This section presents a performance analysis of the proposed methods for the case of 1-bit ADCs. The analysis is applicable for any blind detection scheme for MIMO receivers with low-resolution ADCs and for Rayleigh fading channels, independent of the channel realization. We assume that all symbol vectors in $\check{\mathcal{X}}$ are a priori equally likely to be transmitted. The objective is to characterize the VER. Since the performance of the proposed methods for 1-bit ADCs is independent of the step size Δ, we choose $\Delta = 2$ so that the quantization function becomes the sign$(\cdot)$ function, where sign$(a) = +1$ if $a \geq 0$ and sign$(a) = -1$ if $a < 0$. If a is a complex number, then $\text{sign}(a) = \text{sign}(\Re\{a\}) + j\,\text{sign}(\Im\{a\})$. The operator sign$(\cdot)$ of a matrix or vector is applied separately to every element of that matrix or vector.

3.5.1 VER Analysis at Low SNRs

Here, an approximate pairwise VER at low SNRs for the Rayleigh fading channel is presented. First, using the Bussgang decomposition, the system model $\mathbf{y} = Q_b(\mathbf{r})$ can be rewritten as $\mathbf{y} = \mathbf{V}\mathbf{r} + \mathbf{d}$ [20] where $\mathbf{d}$ is the quantization distortion and

$$\mathbf{V} = \sqrt{\frac{2}{\pi}}\,\text{diag}(\mathbf{\Sigma}_r)^{-\frac{1}{2}}. \tag{3.25}$$

The term $\mathbf{\Sigma}_{\mathbf{r}} = \mathbf{H}\mathbf{H}^H + N_0\mathbf{I}$ is the covariance matrix of $\mathbf{r}$. Let $\mathbf{A} = \mathbf{V}\mathbf{H}$ and $\mathbf{e} = \mathbf{V}\mathbf{z} + \mathbf{d}$, then the system model becomes

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}, \tag{3.26}$$

where $\mathbf{A} = \sqrt{2/\pi}\,\mathrm{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}}\mathbf{H}$ and the effective noise $\mathbf{e} = [e_1, e_2, \ldots, e_{N_{\mathrm{rx}}}]^T$ is modeled as Gaussian [20] with zero mean and covariance matrix

$$\mathbf{\Sigma_e} = \frac{2}{\pi}\Big[\arcsin\Big(\mathrm{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}}\mathbf{\Sigma_r}\,\mathrm{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}}\Big) - \mathrm{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}}\mathbf{\Sigma_r}\,\mathrm{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}} + N_0\,\mathrm{diag}(\mathbf{\Sigma_r})^{-1}\Big]. \tag{3.27}$$

Note that the operation $\arcsin(\cdot)$ of a matrix is applied element-wise on that matrix. The representative vector $\check{\mathbf{y}}_k$ now becomes $\check{\mathbf{y}}_k = \mathbf{A}\check{\mathbf{x}}_k$.

In the low SNR regime, the approximation $\mathbf{\Sigma_r} \approx \mathbf{\Sigma_z}$ holds [20], where $\mathbf{\Sigma_z} = N_0\mathbf{I}$ is the covariance matrix of $\mathbf{z}$. This approximation leads to $\mathbf{A} \approx \sqrt{2/(N_0\pi)}\mathbf{H}$ and $\mathbf{\Sigma_e} \approx \mathbf{I}$. Let $\boldsymbol{\upsilon} = [\upsilon_1, \ldots, \upsilon_{N_{\mathrm{r}}}]^T = \check{\mathbf{y}}_k - \check{\mathbf{y}}_{k'}$, where $\upsilon_i = \sqrt{2/(N_0\pi)}\mathbf{h}_i^T(\check{\mathbf{x}}_k - \check{\mathbf{x}}_{k'})$ with $\mathbf{h}_i$ being the i^{th} column of $\mathbf{H}$. Since $\mathbf{H}$ is comprised of i.i.d. Gaussian random variables $\mathcal{CN}(0,1)$, υ_i is also Gaussian of zero mean with variance

$$\sigma_{kk'}^2 = \frac{2}{N_0\pi}\|\check{\mathbf{x}}_k - \check{\mathbf{x}}_{k'}\|_2^2. \tag{3.28}$$

Denote $P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}}$ as the pairwise vector error probability of confusing $\check{\mathbf{x}}_k$ with $\check{\mathbf{x}}_{k'}$ when $\check{\mathbf{x}}_k$ is transmitted and when $\check{\mathbf{x}}_k$ and $\check{\mathbf{x}}_{k'}$ are the only two hypotheses [2]. The following proposition establishes the relationship between $P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}}$ and $\sigma_{kk'}^2$.

Proposition 3.2. *$P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}}$ at low SNR can be approximated as*

$$P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}} \approx 1 - \Phi\Big(\sqrt{N_{\mathrm{rx}}/(1 + 2/\sigma_{kk'}^2)}\Big). \tag{3.29}$$

Proof. Please refer to Appendix A. □

The result in Proposition 3.2 clearly shows the dependency of the pairwise VER on the Euclidean distance between the two symbol vectors $\check{\mathbf{x}}_k$ and $\check{\mathbf{x}}_{k'}$. We now proceed to obtain an upper bound on the VER, denoted as $P_{\varrho}^{\mathrm{ver}}$, at low SNR assuming a priori equally likely

$\check{\mathbf{x}}_1, \ldots, \check{\mathbf{x}}_K$. The VER is defined as

$$P_\varrho^{\text{ver}} = \sum_{k=1}^{K} \mathbb{P}[\hat{\mathbf{x}} \neq \check{\mathbf{x}}_k, \mathbf{x} = \check{\mathbf{x}}_k]$$

where $\hat{\mathbf{x}}$ is the detected symbol vector and $\mathbb{P}[\hat{\mathbf{x}} \neq \check{\mathbf{x}}_k, \mathbf{x} = \check{\mathbf{x}}_k]$ is the probability that $\check{\mathbf{x}}_k$ was transmitted but the detected symbol vector is not $\check{\mathbf{x}}_k$.

Proposition 3.3. P_ϱ^{ver} *at low SNR is upper-bounded as*

$$P_\varrho^{\text{ver}} \;\; \leq \;\; \frac{1}{K} \sum_{k=1}^{K} \sum_{k' \neq k}^{K} \left[1 - \Phi\Big(\sqrt{N_{\text{rx}}/(1 + 2/\sigma_{kk'}^2)} \Big) \right]. \tag{3.30}$$

Proof. The bound on P_ϱ^{ver} is obtained via the union bound

$$P_\varrho^{\text{ver}} = \sum_{k=1}^{K} \mathbb{P}[\hat{\mathbf{x}} \neq \check{\mathbf{x}}_k, \mathbf{x} = \check{\mathbf{x}}_k] = \frac{1}{K} \sum_{k=1}^{K} \mathbb{P}[\hat{\mathbf{x}} \neq \check{\mathbf{x}}_k \mid \mathbf{x} = \check{\mathbf{x}}_k] \leq \frac{1}{K} \sum_{k=1}^{K} \sum_{k' \neq k}^{K} P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}}$$

and the application of Proposition 3.2. □

The probability $\mathbb{P}[\hat{\mathbf{x}} \neq \check{\mathbf{x}}_k \,|\, \mathbf{x} = \check{\mathbf{x}}_k]$ is invariant to $\check{\mathbf{x}}_k$ for the case of PSK modulation. Without loss of generality, we assume that $\check{\mathbf{x}}_1$ was transmitted, so that the VER simplifies to

$$P_\varrho^{\text{ver}} \leq \sum_{k \neq 1}^{K} \left[1 - \Phi\Big(\sqrt{N_{\text{rx}}/(1 + 2/\sigma_{1k}^2)} \Big) \right]. \tag{3.31}$$

We note that this result is valid for low SNRs. In the following analysis, we characterize the VER at a very high SNR, i.e., $\varrho \to \infty$.

3.5.2 VER Analysis as SNR $\to \infty$

Here, the VER as $\varrho \to \infty$ is evaluated. Let $\mathbf{g}_k = [g_{k,1}, \ldots, g_{k,N_{\text{rx}}}]^T = \mathbf{H}\check{\mathbf{x}}_k$, then

$$\mathbb{P}[\Re\{y_i\} = +1 \mid \mathbf{x} = \check{\mathbf{x}}_k] = \Phi(\sqrt{2\varrho/N_{\text{tx}}}\,\Re\{g_{k,i}\}), \tag{3.32}$$

$$\mathbb{P}[\Im\{y_i\} = +1 \mid \mathbf{x} = \check{\mathbf{x}}_k] = \Phi(\sqrt{2\varrho/N_{\text{tx}}}\,\Im\{g_{k,i}\}). \tag{3.33}$$

The true representative vectors are

$$\check{\mathbf{y}}_k = \mathbb{E}\big[\mathbf{y} \mid \mathbf{x} = \check{\mathbf{x}}_k\big] = 2\Phi(\sqrt{2\varrho/N_{\text{tx}}}\mathbf{g}_k) - (\mathbf{1} + j\mathbf{1}) \tag{3.34}$$

which becomes $\operatorname{sign}(\mathbf{g}_k)$ as $\varrho \to \infty$. It is possible for a given realization of $\mathbf{H}$ that more than one symbol vector will lead to the same representative vector: $\operatorname{sign}(\mathbf{g}_k) = \operatorname{sign}(\mathbf{g}_{k'})$ with $k \neq k'$, and in such cases a detection error will occur regardless of the detection scheme. In the following, we analyze the probability that $\operatorname{sign}(\mathbf{g}_k) = \operatorname{sign}(\mathbf{g}_{k'})$. The analysis is applicable for the cases of BPSK and QPSK modulation.

To facilitate the analysis, we convert the notation into the real domain as follows:

$$\check{\mathbf{x}}_k^{\mathbb{R}} = [\check{x}_{k,1}^{\mathbb{R}}, \check{x}_{k,2}^{\mathbb{R}}, \ldots, \check{x}_{k,2N_{\text{tx}}}^{\mathbb{R}}]^T = [\Re\{\check{\mathbf{x}}_k\}^T, \Im\{\check{\mathbf{x}}_k\}^T]^T,$$

$$\mathbf{g}_k^{\mathbb{R}} = [g_{k,1}^{\mathbb{R}}, g_{k,2}^{\mathbb{R}}, \ldots, g_{k,2N_{\text{rx}}}^{\mathbb{R}}]^T = [\Re\{\mathbf{g}_k\}^T, \Im\{\mathbf{g}_k\}^T]^T.$$

We first consider BPSK modulation, i.e., $\mathcal{M} = \{\pm 1\}$. In this case, $\Im\{\check{\mathbf{x}}_k\} = \mathbf{0}$.

Theorem 3.1. *Given $d = \|\check{\mathbf{x}}_k^{\mathbb{R}} - \check{\mathbf{x}}_{k'}^{\mathbb{R}}\|_0$ as the Hamming distance between the two labels, then*

$$\mathbb{P}\big[\operatorname{sign}(\mathbf{g}_k) = \operatorname{sign}(\mathbf{g}_{k'})\big] = \left[\frac{2}{\pi}\arctan\sqrt{\frac{N_{\text{tx}} - d}{d}}\right]^{2N_{\text{rx}}}. \tag{3.35}$$

Proof. Please refer to Appendix B. □

As $\varrho \to \infty$, the effect of additive white Gaussian noise (AWGN) can be ignored. Thus, $\mathbb{P}\big[\check{\mathbf{y}}_k = \check{\mathbf{y}}_{k'}\big] = \mathbb{P}\big[\mathrm{sign}(\mathbf{g}_k) = \mathrm{sign}(\mathbf{g}_{k'})\big]$. An upper bound on the VER is established in the following proposition.

Proposition 3.4. *With BPSK modulation, the asymptotic VER at high SNR is upper-bounded as*

$$P^{\mathrm{ver}}_{\varrho\to\infty} \leq \frac{1}{2}\sum_{d=1}^{N_{\mathrm{tx}}} \binom{N_{\mathrm{tx}}}{d} \left[\frac{2}{\pi}\arctan\sqrt{\frac{N_{\mathrm{tx}}-d}{d}}\right]^{2N_{\mathrm{rx}}}. \tag{3.36}$$

Proof. Please refer to Appendix C. □

Proposition 3.5. *With BPSK modulation and $N_{\mathrm{tx}} = 2$, the upper bound in* (3.36) *is tight.*

Proof. For BPSK modulation and $N_{\mathrm{tx}} = 2$, let $\check{\mathbf{x}}_1^{\mathbb{R}} = [1, 1, 0, 0]$, $\check{\mathbf{x}}_2^{\mathbb{R}} = [1, -1, 0, 0]$, $\check{\mathbf{x}}_3^{\mathbb{R}} = [-1, 1, 0, 0]$, $\check{\mathbf{x}}_4^{\mathbb{R}} = [-1, -1, 0, 0]$. Herein, $\check{\mathbf{x}}_1^{\mathbb{R}} = -\check{\mathbf{x}}_4^{\mathbb{R}}$ and $\check{\mathbf{x}}_2^{\mathbb{R}} = -\check{\mathbf{x}}_3^{\mathbb{R}}$, resulting in $\check{\mathbf{y}}_1 = -\check{\mathbf{y}}_4$ and $\check{\mathbf{y}}_2 = -\check{\mathbf{y}}_3$ as $\varrho \to \infty$. Hence, events $\check{\mathbf{y}}_1 = \check{\mathbf{y}}_2$ and $\check{\mathbf{y}}_1 = \check{\mathbf{y}}_3$ are mutually exclusive while event $\check{\mathbf{y}}_1 = \check{\mathbf{y}}_4$ does not exist. This proposition thus follows as a direct consequence of the proof for Proposition 3.4 given in Appendix C. □

For the case of QPSK modulation, the Hamming distance $d = \|\check{\mathbf{x}}_k^{\mathbb{R}} - \check{\mathbf{x}}_{k'}^{\mathbb{R}}\|_0$ between any two labels can be as large as $2N_{\mathrm{tx}}$. Following the same derivation as in Theorem 3.1 and Proposition 3.4, an upper-bound for the asymptotic VER at high SNR can be established by the following proposition.

Proposition 3.6. *With QPSK modulation, the asymptotic VER at high SNR is upper-bounded as*

$$P^{\mathrm{ver}}_{\varrho\to\infty} \leq \frac{1}{2}\sum_{d=1}^{2N_{\mathrm{tx}}} \binom{2N_{\mathrm{tx}}}{d} \left[\frac{2}{\pi}\arctan\sqrt{\frac{2N_{\mathrm{tx}}-d}{d}}\right]^{2N_{\mathrm{rx}}}. \tag{3.37}$$

3.5.3 Transmit Signal Design

Thus far it has been assumed that the transmitter uses all K possible labels for transmission. However, as K grows large, the training task for all the K labels becomes impractical, since the block fading interval T_{b} is finite. In this section, we consider a system where the transmitter employs only a subset of $\tilde{K}$ labels among the K possible labels for both the training and data transmission phases. The rest of the $K - \tilde{K}$ labels are unused. While using only $\tilde{K}$ labels reduces the transmission rate as compared to using all the K possible labels, the VER can be improved. In many 5G networks, e.g., Machine-to-Machine (M2M) communication systems, the priority is on the reliability, not the rate [6]. In addition, the reduction in training time with small $\tilde{K}$ may help improve the system throughput.

The design problem is how to choose $\tilde{K}$ labels among the K labels. To address this problem, we rely on Proposition 3.4 and Proposition 3.6. These propositions reveal that the VER at infinite SNR is inversely proportional to the Hamming distances between the labels. Thus, the following criterion for choosing the transmit signals is proposed:

$$\mathcal{X}^{\star} = \arg\max_{\mathcal{X} \subset \check{\mathcal{X}}^{\mathbb{R}}} \min_{1 \le k_1 < k_2 \le \tilde{K}} \|\mathbf{x}_{k_1} - \mathbf{x}_{k_2}\|_0, \tag{3.38}$$

where $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{\tilde{K}}\}$ denote the set of $\tilde{K}$ different labels for transmission, and $\check{\mathcal{X}}^{\mathbb{R}} = \{\check{\mathbf{x}}_1^{\mathbb{R}}, \ldots, \check{\mathbf{x}}_K^{\mathbb{R}}\}$. This design criterion aims to maximize the minimum pairwise Hamming distance among the $\tilde{K}$ labels. Note that the proposed criterion is also applicable for low SNRs because as shown in Proposition 3.3, the VER is inversely proportional to the Euclidean distance, which is analogous to the Hamming distance for BPSK and QPSK, albeit with some scaling factor. It should be noted that the proposed criterion does not rely on a specific channel realization, so the design task can be carried out off-line.

Problem (3.38) can be solved by exhaustive search when $\binom{K}{\tilde{K}}$ is not too large. When the exhaustive search is not possible, we propose a simple greedy algorithm, whose pseudo-

Algorithm 3: Transmit Signal Design.

```
1  Randomly generate $N_{\text{set}}$ initial sets $\{\mathcal{X}_i, i = 1, \dots, N_{\text{set}}\}$;
2  for $i = 1 : N_{\text{set}}$ do
3      $done = false$;
4      while $done$ = $false$ do
5          Let $flag = 1$;
6          Set $\mathcal{X}' = \check{\mathcal{X}} \backslash \mathcal{X}_i = \{\mathbf{x}'_1, \dots, \mathbf{x}'_{K-\tilde{K}}\}$;
7          for $k_1 = 1 : \tilde{K}$ do
8              for $k_2 = 1 : K - \tilde{K}$ do
9                  Let $\hat{\mathcal{X}}_i = \big(\mathcal{X}_i \backslash \{\mathbf{x}_{k_1}\}\big) \cup \{\mathbf{x}'_{k_2}\}$;
10                 if $d_{\min}(\hat{\mathcal{X}}_i) > d_{\min}(\mathcal{X}_i)$ then
11                     Set $\mathcal{X}_i = \hat{\mathcal{X}}$ and $flag = 0$;
12                     Exit both for loops;
13                 end
14             end
15         end
16         if $flag = 1$ then
17             Set $done = true$ and $\mathcal{X}_i^\star = \mathcal{X}_i$;
18         end
19     end
20 end
21 $\mathcal{X}^\star = \arg\max_{\mathcal{X}_i^\star} d_{\min}(\mathcal{X}_i^\star)$;
```

code can be found in Algorithm 3. Here, $d_{\min}(\mathcal{X})$ denotes the minimum pairwise Hamming distance among the labels in $\mathcal{X}$ and $\mathcal{X}'$ in line 6 denotes the set of labels, which is not used for transmission. The principle of Algorithm 3 is as follows:

- Generate N_{set} initial sets $\{\mathcal{X}_i\}_{i=1,\dots,N_{\text{set}}}$, where each set $\mathcal{X}_i$ contains $\tilde{K}$ different labels randomly chosen from $\check{\mathcal{X}}^{\mathbb{R}}$.

- For each initial set $\mathcal{X}_i$, find $\mathbf{x}' \in \mathcal{X}'$ such that when an element of $\mathcal{X}_i$ is replaced by $\mathbf{x}'$, the value of the objective function in (3.38), i.e., the minimum Hamming distance, is increased. This is repeated until no further increase in the objective function is possible after evaluating all replacements.

- Each initial set $\mathcal{X}_i$ produces a corresponding solution $\mathcal{X}_i^\star$ as in line 17. The solution $\mathcal{X}^\star$ of (3.38) is obtained by selecting the $\mathcal{X}_i^\star$ whose objective function value is largest (line 21).

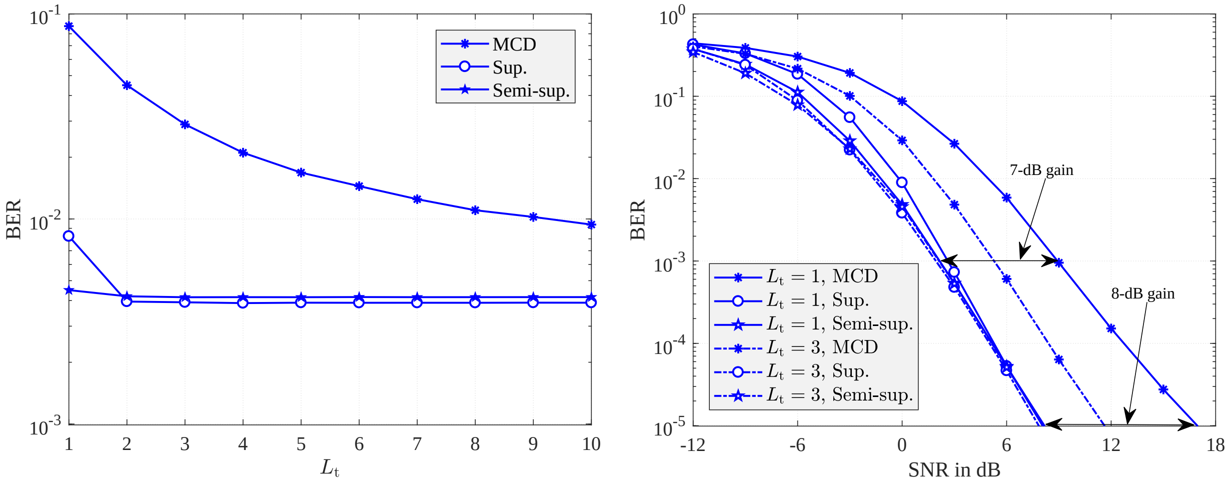
Note that the larger N_{set} is, the more likely Algorithm 3 will find the optimal solution.

3.6 Simulations and Results

3.6.1 Numerical Evaluation of the Proposed Methods

Monte Carlo simulations are used to numerically evaluate the performance of the proposed methods. The simulation settings are as follows. The number of transmit antennas N_{tx} is set to be 2 unless otherwise stated. The data phase contains $T_{\text{d}} = 500$ time slots. In the supervised learning method, a 24-bit CRC as in the 3GPP Long Term Evolution (LTE) standard [85] is adopted. The generator of the CRC in the simulation is $z^{24}+z^{23}+z^{14}+z^{12}+z^{8}+1$, and the length of each data segment is 16 bits. Thus, the length of each coded segment is 40 bits. This is the minimum length in the 3GPP LTE standard. In all figures, 'Sup.' and 'Semi-sup.' stand for the supervised learning and semi-supervised learning methods, respectively.

The effect of the training sequence length L_{t} on MCD and the two proposed methods is first studied (Figure 3.3). BPSK modulation with $N_{\text{rx}} = 16$ and 1-bit ADCs are used. Figure 3.3a shows the change of the BER as L_{t} varies. An interesting observation is that the performance of the proposed methods is much less affected by L_{t} than the MCD method. Hence, the length of the training sequence can be reduced without causing much degradation on the performances of the proposed methods. This is illustrated more clearly in Figure 3.3b, where we carry out the simulation for $L_{\text{t}} = 1$ and $L_{\text{t}} = 3$, still with BPSK modulation, 1-bit ADCs and $N_{\text{rx}} = 16$. It can be seen from Figure 3.3b that, as L_{t} is reduced from 3 to 1, the BER of MCD is significantly degraded while the BERs of the proposed methods experience only a



(a) L_t varies and $\varrho = 0$ dB.

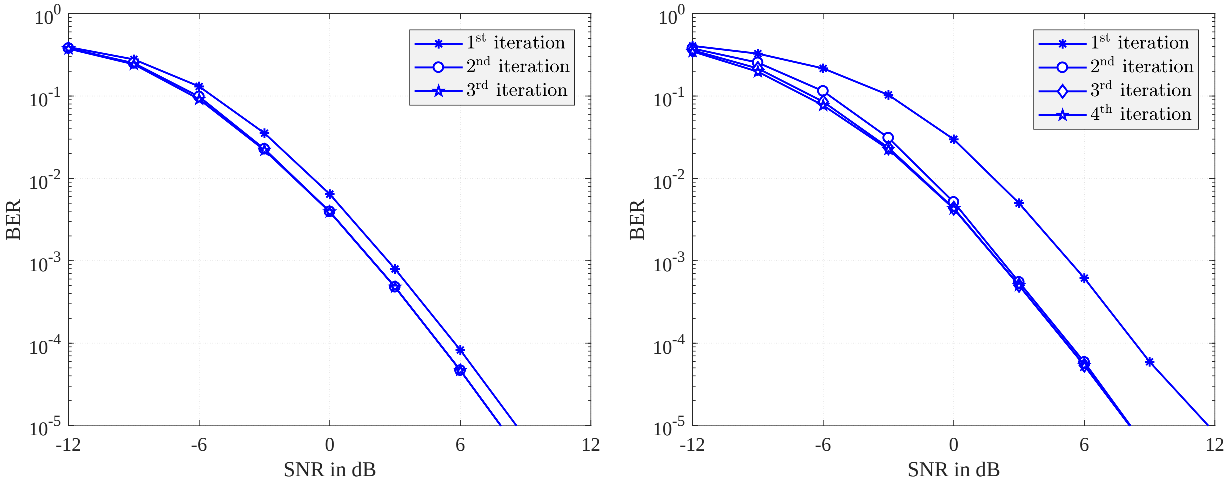
(b) $L_t = 1$ and $L_t = 3$, ϱ varies.

Figure 3.3: Effect of L_t on MCD and the proposed methods with 1-bit ADCs, $N_{rx} = 16$ and BPSK modulation.

small degradation at low SNRs and do not change at higher SNRs. This leads to a significant improvement for the proposed methods as compared to MCD, for example, about a 7-dB gain at a BER of 10^{-3} and 8-dB at a BER of 10^{-5} when $L_t = 1$. Even for moderately long training sequences, e.g., $L_t = 3$, the gain of the proposed methods is still considerable, from 3-dB to 4-dB.

The results in Figure 3.3 can be explained as follows. The performance of MCD is susceptible to L_t because its detection accuracy relies on the representative vectors estimated only from the training sequence. Therefore, if L_t is small, the representative vectors are not estimated correctly and so the performance can be degraded significantly. On the other hand, the proposed methods are much less dependent on L_t because they use the training sequence only as the initial guide for the detection task. Compared to the semi-supervised learning method, the supervised learning method is slightly more dependent on L_t because it depends on detection results from the training sequence.

Since the proposed methods work iteratively, numerous simulations are performed to evaluate the improvement in BER over the iterations. Simulation results are shown in Figure 3.4. For



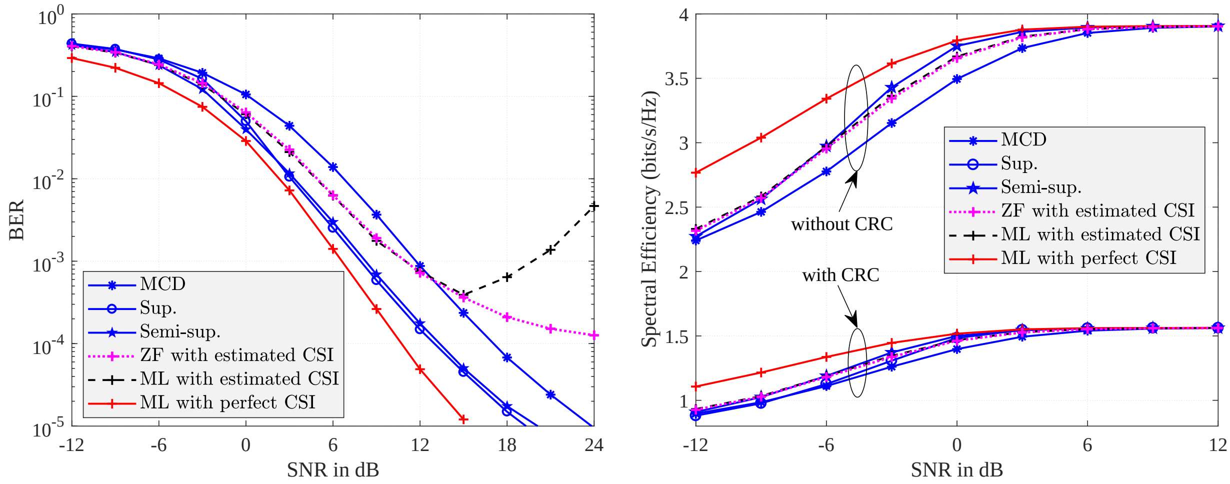
(a) Supervised learning method.

(b) Semi-supervised learning method.

Figure 3.4: Performance improvement for different iterations with 1-bit ADCs, BPSK modulation, $N_{\mathrm{rx}} = 16$ and $L_{\mathrm{t}} = 3$.

the supervised learning method, Figure 3.4a, it can be seen that the BER converges after only 2 iterations. For the semi-supervised learning method, Figure 3.4b, there is considerable improvement between the first and the second iterations, but then the third and the fourth iterations give approximately the same performance. It is therefore preferred to limit the maximum number of iterations to 3 in the semi-supervised learning method. It should be noted that the BER on the first iteration of the semi-supervised learning method is actually the BER of the MCD method because the first iteration only exploits the training sequence.

Figure 3.5 compares the aforementioned blind detection methods with several coherent detection methods. The simulation uses 1-bit ADCs, QPSK modulation, $N_{\mathrm{rx}} = 16$ and $L_{\mathrm{t}} = 3$. For coherent detection, CSI is first estimated by the Bussgang Linear Minimum Mean Squared Error (BLMMSE) method proposed in [33]. The length of the training sequence in the blind detection methods is 12, so we also set the length of the pilot sequence for the channel estimation to 12. The ZF detection method is presented in [33]. The ML method for 1-bit ADCs is provided in [31]. A performance comparison in terms of BER is given in Figure 3.5a, which shows that the proposed methods outperform the ZF and ML methods with estimated CSI. It is also seen that the BER of the proposed methods is quite close the BER of ML detection



(a) BER

(b) Spectral efficiency

Figure 3.5: Performance comparison between blind and coherent detection with 1-bit ADCs, QPSK modulation, $N_{\mathrm{rx}} = 16$ and $L_{\mathrm{t}} = 3$.

with perfect CSI. Here, it is observed that a significant increase in the BER at high SNRs for the ML method with estimated CSI. This observation was also reported in [79]. In comparing the two proposed methods in Figure 3.5a and Figure 3.3, should the CRC be available, it is more beneficial to use the supervised learning method for better BER performance.

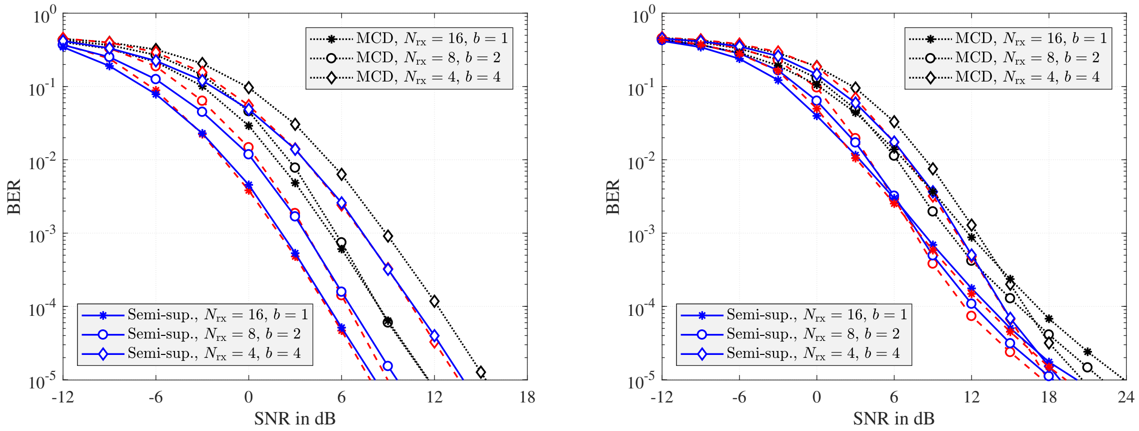
Figure 3.5b provides a comparison in terms of spectral efficiency η_{se}, defined as the average number of information bits received correctly per block-fading interval T_{b}. We determine η_{se} for the case without CRC as

$$\eta_{\mathrm{se}} = \frac{T_{\mathrm{d}}}{T_{\mathrm{b}}} \times (1 - \mathrm{BER}) \times N_{\mathrm{tx}} \times \log_2 M$$

and for the case with CRC as

$$\eta_{\mathrm{se}} = \frac{L_{\mathrm{data}}}{L_{\mathrm{data}} + L_{\mathrm{CRC}}} \times \frac{T_{\mathrm{d}}}{T_{\mathrm{b}}} \times (1 - \mathrm{BER}) \times N_{\mathrm{tx}} \times \log_2 M.$$

Figure 3.5b indicates a proportional drop in the spectral efficiency due to the use of CRC. Note that the supervised learning method can only be applied in systems where the CRC



(a) BPSK modulation.

(b) QPSK modulation.

Figure 3.6: Performance of the proposed methods for different numbers of receive antennas N_{rx} and ADC resolutions b with $L_{\mathrm{t}} = 3$.

is available but the other methods can be used in any system regardless of the CRC. Thus, should the CRC be eliminated for improved spectral efficiency, the semi-supervised method provides better performance than MCD. It also performs slightly better than conventional coherent detection with estimated CSI. The small performance gap observed in Figure 3.5b is due to the small difference in BER performance in the SNR region between -12 to 12 dB, as shown in Figure 3.5a. At high SNR, while the proposed method performs much better than other methods in terms of BER, its effect on the throughput η_{se} is negligible since $1 - \mathrm{BER} \approx 1$.

To study the trade-off between N_{rx} and b, the proposed methods are evaluated in three different scenarios: (i) $N_{\mathrm{rx}} = 4, b = 4$; (ii) $N_{\mathrm{rx}} = 8, b = 2$; and (iii) $N_{\mathrm{rx}} = 16, b = 1$. This is to ensure the same number of bits after the ADCs for baseband processing. The number of label repetitions L_{t} is set to be 3. The simulation results are shown in Figure 3.6, with BPSK in Figure 3.6a and QPSK in Figure 3.6b. For BPSK modulation, the best performance is achieved by scenario (iii) for all methods. Hence, this suggests the use of more receive antennas and fewer bits in the ADCs when BPSK modulation is employed. However, for QPSK modulation, there is a trade-off between scenarios (ii) and (iii). For low SNRs, the setting $N_{\mathrm{rx}} = 16$ and $b = 1$ gives better performance, but for high SNRs, the best results

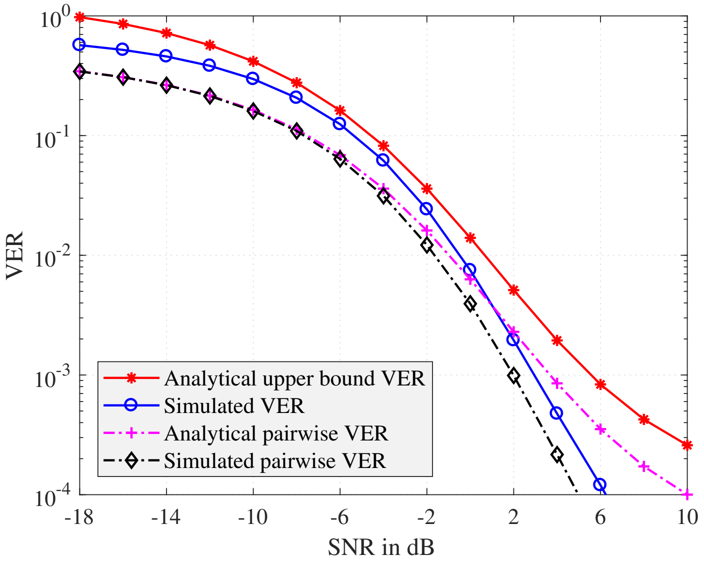


Figure 3.7: Validation of the analytical pairwise VER in (3.29) and the analytical VER in (3.30) at low SNRs with $N_{\text{tx}} = 2$, $N_{\text{rx}} = 16$, and BPSK modulation.

are with $N_{\text{rx}} = 8$ and $b = 2$. The results in Figure 3.6 also show that the proposed methods outperform the MCD method in all three scenarios.

3.6.2 Validation of Performance Analysis

This section presents a validation on the performance analyses in Section 3.5. Figure 3.7 provides the analytical approximate pairwise VER in (3.29) and the VER in (3.30). the setting of $N_{\text{tx}} = 2$, $N_{\text{rx}} = 16$, and BPSK modulation is used. The two labels used to examine the pairwise VER are $\check{\mathbf{x}}_k = [+1, +1]^T$ and $\check{\mathbf{x}}_{k'} = [+1, -1]^T$. It can be seen that our approximate pairwise VER is very close to the simulated pairwise VER at low SNRs, typically with SNRs less than 0-dB. However, as the SNR increases, the approximate pairwise VER tends to diverge from the true pairwise VER because the approximation $\boldsymbol{\Sigma}_{\mathbf{r}} \approx \boldsymbol{\Sigma}_{\mathbf{z}}$ is inapplicable for high SNRs. The simulation results also show that the analytical VER is quite close to the true VER at low SNRs.

Validation of the high SNR expressions for the analytical VER is given in Figure 3.8 with

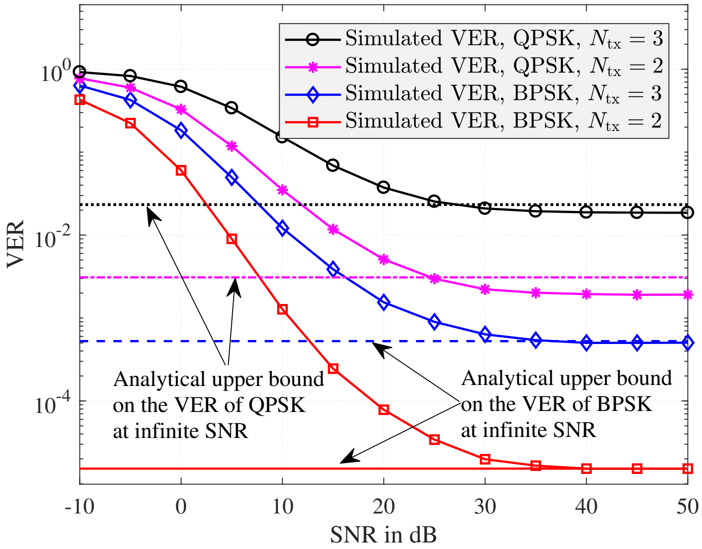


Figure 3.8: Validation of the analytical VER at infinite SNR in Propositions 3.4, 3.5, and 3.6.

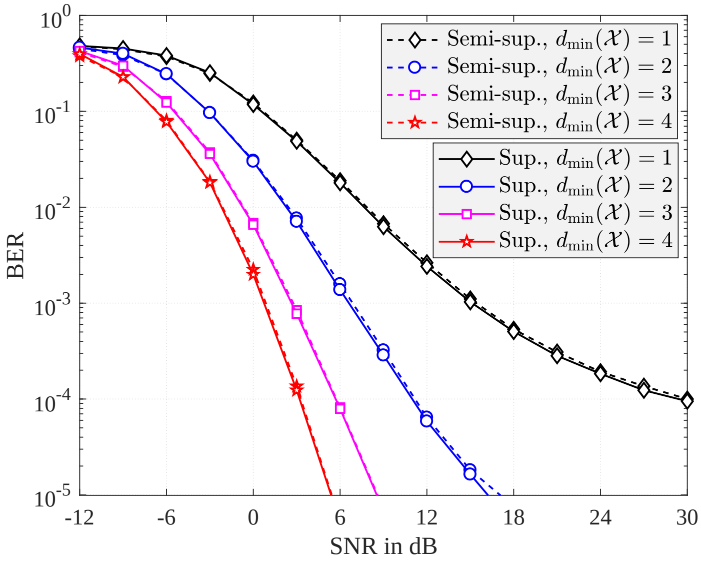


Figure 3.9: Validation of the transmit signal design with $N_{\text{tx}} = 6$, $N_{\text{rx}} = 16$, $\tilde{K} = 4$, and BPSK modulation.

$N_{\text{rx}} = 8$. The horizontal lines represent the analytical upper bounds on the VER at infinite SNR. For the case of BPSK and $N_{\text{tx}} = 2$, it can be seen that the simulated VER approaches the horizontal solid line as the SNR increases and then they match at very high SNRs. This validates the result of Proposition 3.5 indicating that the bound is tight in the case of BPSK and $N_{\text{tx}} = 2$. With BPSK and $N_{\text{tx}} = 3$, the horizontal dashed line is just slightly higher than the floor of the simulated VER. For QPSK modulation, there is a small gap between the horizontal lines and the floors of the simulated VER. These observations validate the analytical upper-bound results in Proposition 3.4 and Proposition 3.6.

Figure 3.9 provides a validation for the proposed transmit signal design based on the minimum Hamming distance in Section 3.5.3. With different selections of the label sets $\mathcal{X}$, the BER performance in Figure 3.9 improves as $d_{\min}(\mathcal{X})$ increases, which validates the analysis. In this particular simulation scenario ($N_{\text{tx}} = 6$, $N_{\text{rx}} = 16$, $\tilde{K} = 4$, and BPSK modulation), the minimum Hamming distance of an optimal set can be found to be 4. The proposed Algorithm 3 then helps select an optimal set $\mathcal{X}^{\star}$ with $d_{\min}(\mathcal{X}^{\star}) = 4$. Hence, the curves with star markers in Figure 3.9 also represent the BER obtained by $\mathcal{X}^{\star}$.

As $\tilde{K}$ is increased, the data rate also increases, but the BER will degrade. Thus, there is a specific value for $\tilde{K}$ that provides the best compromise for the spectral efficiency. Figure 3.10 illustrates the change of spectral efficiency with respect to $\tilde{K}$ at different SNR values. The simulations are carried out with $N_{\text{tx}} = 8$, $N_{\text{rx}} = 16$, QPSK modulation, $L_{\text{t}} = 3$, and $\tilde{K} \in \{4, 8, 16, 32, 64, 128\}$. The maximum number of time slots for the block-fading interval is $T_{\text{b}} = 500$. The availability of the CRC is assumed so that the supervised learning method can be compared with other methods. The lengths of the data segment for $\tilde{K} \in \{4, 8, 64, 128\}$ and $\tilde{K} \in \{16, 32\}$ are 18 bits and 16 bits, respectively. This is to ensure that the number of bits in a segment is a multiple of the number bits in a transmitted vector. The length of the data block T_{d} is also set to be a multiple of $(L_{\text{CRC}} + L_{\text{data}})/\log_2 \tilde{K}$. The spectral efficiency

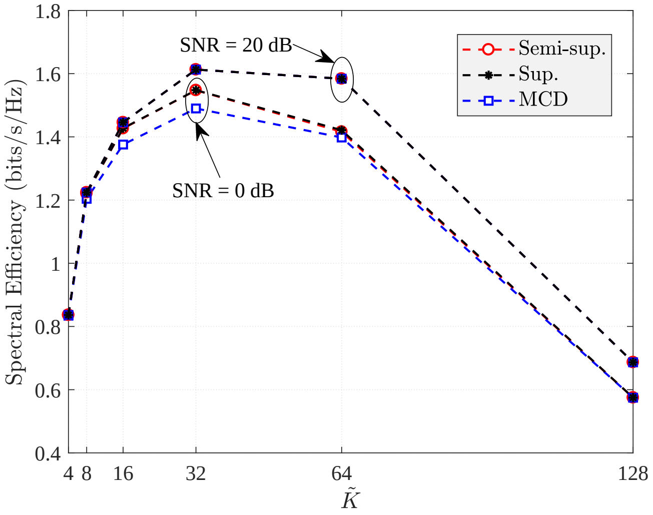


Figure 3.10: Spectral efficiency versus $\tilde{K}$ with $N_{\mathrm{tx}} = 8$, $N_{\mathrm{rx}} = 16$, QPSK modulation, $L_{\mathrm{t}} = 3$, and $T_{\mathrm{b}} = 500$.

is then computed as

$$\eta_{\mathrm{se}} = \frac{L_{\mathrm{data}}}{L_{\mathrm{CRC}} + L_{\mathrm{data}}} \times \frac{T_{\mathrm{d}}}{T_{\mathrm{d}} + T_{\mathrm{t}}} \times (1 - \mathrm{BER}) \times \log_2 \tilde{K}.$$

For each value of $\tilde{K}$, Algorithm 3 is applied to find the solution $\mathcal{X}^*$ of (3.38). It is found that the symbol vectors of $\mathcal{X}^*$ do not satisfy Condition 2, and so the full-space training method is used. The simulation results in Figure 3.10 show that increasing $\tilde{K}$ does not necessarily improve the spectral efficiency, due to the increased training overhead. There is thus an optimal value of $\tilde{K} = 32$ in this scenario. It is also seen that at low SNR the spectral efficiencies of the proposed methods are higher than that of MCD.

3.7 Conclusion

In this chapter, blind detection in MIMO systems with low-resolution ADCs is studied. Two new learning methods for enhancing the detection performance were proposed. While the supervised learning method exploits the use of CRC in practical systems to gain more training data, the semi-supervised learning method is based on the perspective that the to-be-decoded data can itself help the detection task thanks to grouping of received symbol vectors for the same transmitted signal. Simulation results demonstrate the performance improvement and robustness of our proposed methods over existing techniques. Numerical results also show that the two proposed learning methods require only a few iterations to converge. We have also carried out a performance analysis for the proposed methods by evaluating the VER in different SNR regimes. In addition, a new criterion for the transmit signal design problem has also been proposed.

Chapter 4

SVM-based channel estimation and data detection for massive MIMO systems with one-bit ADCs

4.1 Introduction

In this chapter, we propose channel estimation and data detection methods for massive MIMO systems with 1-bit ADCs. The proposed methods are efficient, robust, and applicable to large-scale systems without the need for CRC or error correcting codes. This work is based on SVM, a well-known supervised-learning technique in machine learning [84]. Since SVM problems can be solved by very efficient algorithms [88–92], the proposed methods can be

The materials presented in Chapter 4 have been presented at the 2020 IEEE International Conference on Communications (ICC) in Dublin, Ireland [86] and published in the IEEE Transactions on Signal Processing [87] implemented in an efficient manner. There are several prior works on the application of SVM to channel estimation and data detection problems, e.g., [93,94]. However, these works consider either SISO or SIMO channels with full-resolution ADCs. In this chaper, we focus on massive MIMO with one-bit ADCs where both i.i.d. and spatially correlated fading channels are considered. The contributions of this chapter are summarized as follows:

- An SVM-based channel estimation method for uncorrelated channels is first proposed by formulating the 1-bit ADC channel estimation problem as an SVM problem. Unlike the soft-SVM method in [37], the proposed method exploits the original idea of SVM by maximizing the margin achieved by the linear discriminator. For spatially correlated channels, we develop a new channel estimation problem by revising the conventional SVM objective function. Numerical results show that the high-SNR Normalized Mean-Squared Error (NMSE) floor of the proposed channel estimation methods is lower than that of the BMMSE method proposed in [33], which outperforms other existing methods.

- We then propose a two-stage SVM-based data detection method, where the first stage is also formulated as an SVM problem. A second stage is then employed to refine the solution from the first stage. Simulation results show that the performance of the proposed method is very close to that of the ML detection method if perfect CSI is available. With imperfect CSI, the proposed data detection method is shown to be robust and to also outperform existing methods. We then consider an SVM-based joint Channel Estimation and Data Detection (CE-DD) method where the to-be-decoded data vectors and pilot data vectors are both exploited to refine the estimated channel and thus improve the data detection performance.

- Finally, an extension of the proposed methods to OFDM systems with frequency-selective fading channels is derived. Numerical results show that the proposed SVM-based methods significantly outperform existing ones. For example, the high-SNR

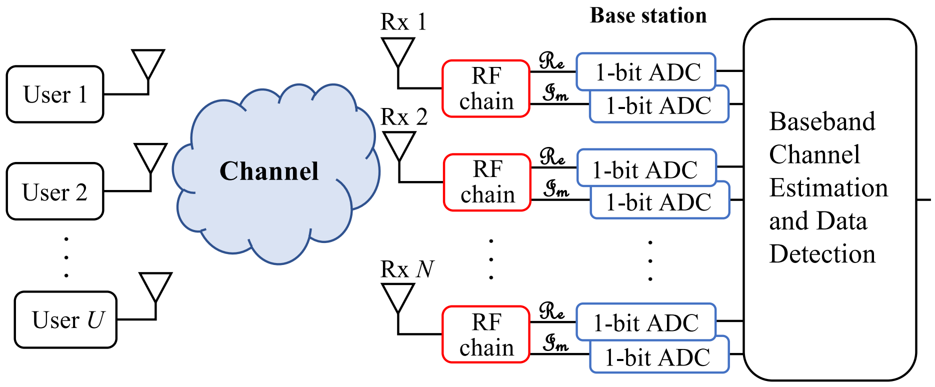


Figure 4.1: Block diagram of a massive MIMO system with U single-antenna users and an N-antenna base station equipped with $2N$ 1-bit ADCs.

NMSE floor of the proposed SVM-based channel estimation method is about 3-dB lower that of the BMMSE method.

The rest of this paper is organized as follows: The system model is first presented in Section 4.2. SVM-based methods for flat-fading channels are then proposed in Section 4.3. Section 4.4 includes an extension of the proposed methods to OFDM sysems with frequency-selective fading channels. Numerical results are provided in Section 4.5 and finally Section 4.6 concludes the chapter.

4.2 System Model

The considered massive MIMO system is illustrated in Figure 4.1 with U single-antenna users and an N-antenna base station, where it is assumed that $N \geq U$. Let $\mathbf{x}^{\mathbb{C}} = [x_1^{\mathbb{C}}, x_2^{\mathbb{C}}, \ldots, x_U^{\mathbb{C}}]^T \in \mathbb{C}^U$ denote the transmitted signal vector, where $x_u^{\mathbb{C}}$ is the signal transmitted from the u^{th} user under the power constraint $\mathbb{E}[|x_u^{\mathbb{C}}|^2] = 1$, $u \in \mathcal{U} = \{1, 2, \ldots, U\}$. Let $\mathbf{H}^{\mathbb{C}} \in \mathbb{C}^{N \times U}$ denote the channel, which for the moment is assumed to be block flat fading. Let $\mathbf{r}^{\mathbb{C}} = [r_1^{\mathbb{C}}, r_2^{\mathbb{C}}, \ldots, r_N^{\mathbb{C}}]^T \in \mathbb{C}^N$ be the unquantized received signal vector at the base station, which is given as

$$\mathbf{r}^{\mathbb{C}} = \mathbf{H}^{\mathbb{C}}\mathbf{x}^{\mathbb{C}} + \mathbf{z}^{\mathbb{C}}, \tag{4.1}$$

where $\mathbf{z}^{\mathbb{C}} = [z_1^{\mathbb{C}}, z_2^{\mathbb{C}}, \ldots, z_N^{\mathbb{C}}]^T \in \mathbb{C}^N$ is a noise vector whose elements are assumed to be i.i.d. as $z_i^{\mathbb{C}} \sim \mathcal{CN}(0, N_0)$, and N_0 is the noise power. Each analog received signal $r_i^{\mathbb{C}}$ is then quantized by a pair of 1-bit ADCs. Hence, we have the received signal

$$\mathbf{y}^{\mathbb{C}} = \text{sign}(\mathbf{r}^{\mathbb{C}}) = \text{sign}\left(\Re\{\mathbf{r}^{\mathbb{C}}\}\right) + j\,\text{sign}\left(\Im\{\mathbf{r}^{\mathbb{C}}\}\right) \tag{4.2}$$

where $\text{sign}(\cdot)$ represents the 1-bit ADC with $\text{sign}(a) = +1$ if $a \geq 0$ and $\text{sign}(a) = -1$ if $a < 0$. The operator $\text{sign}(\cdot)$ of a matrix or vector is applied separately to every element of that matrix or vector. The SNR is defined as $\rho = 1/N_0$.

4.3 Proposed SVM-based Channel Estimation and Data Detection with One-bit ADCs

4.3.1 Introduction to Support Vector Machines

SVMs are a family of supervised learning models often used for classification problems where decision boundaries are found to separate observations in different classes [84]. The original idea of SVM was introduced by Vladimir N. Vapnik and Alexey Ya. Chervonenkis and first published in 1964 within the framework of the "Generalised Portrait Method" for computer learning and pattern recognition [95]. This original SVM algorithm was constructed for classification problems that are linearly separable. In 1995, Corinna Cortes and Vapnik proposed soft-margin SVMs – the commonly-used SVM version today that can deal with non-linearly separable data sets [96]. A mathematical description of SVMs for binary classification problems is given as follows:

Consider a binary classification problem with a training data set of P data pairs $\mathcal{D} = \{(\mathbf{x}_q, y_q)\}_{q=1,\ldots,P}$ where $\mathbf{x}_q$ is a training data point and $y_q \in \{\pm 1\}$ is an associated class label. Note that $\{\mathbf{x}_q\}$ here are vectors of real elements. The data set $\mathcal{D}$ is said to be linearly separable if and only if there exists a linear function $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$ such that $\forall q \in \{1, 2, \ldots, P\}$, $f(\mathbf{x}_q) > 0$ if $y_q = +1$ and $f(\mathbf{x}_q) < 0$ if $y_q = -1$. Here, $\mathbf{w}$ and b are referred to as the weight vector and the bias, respectively. In other words, the hyperplane $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b = 0$ divides the space into two regions where $f(\mathbf{x}) = 0$ acts as the *decision boundary*. The margin of the hyperplane $f(\mathbf{x}) = 0$ with respect to $\mathcal{D}$ is defined as

$$m_{\mathcal{D}}(f) = \frac{2}{\|\mathbf{w}\|}. \tag{4.3}$$

The SVM technique seeks to find $\mathbf{w}$ and b such that the margin $m_{\mathcal{D}}(f)$ is maximized. The optimization problem can be expressed as [84]

$$\begin{aligned} &\underset{\{\mathbf{w},b\}}{\text{minimize}} && \frac{1}{2}\|\mathbf{w}\|^2 \\ &\text{subject to} && y_q(\mathbf{w}^T\mathbf{x}_q + b) \geq 1, \quad q = 1, 2, \ldots, P. \end{aligned} \tag{4.4}$$

In case the training data set $\mathcal{D}$ is not linearly separable, a generalized soft-margin optimization problem is considered as follows [96]:

$$\begin{aligned} &\underset{\{\mathbf{w},b,\xi_q\}}{\text{minimize}} && \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{q=1}^{P} \ell(\xi_q) \\ &\text{subject to} && y_q(\mathbf{w}^T\mathbf{x}_q + b) \geq 1 - \xi_q, \\ &&& \xi_q \geq 0, \quad q = 1, 2, \ldots, P. \end{aligned} \tag{4.5}$$

Here, $\{\xi_q\}$ are slack variables and $C > 0$ is a parameter that “controls the trade-off between the slack variable penalty and the margin” [84], and $\ell(\xi_q)$ is a function of ξ_q. In the SVM literature, two common forms of $\ell(\xi_q)$ are $\ell(\xi_q) = \xi_q$ and $\ell(\xi_q) = \xi_q^2$, which are often referred to as ℓ_1-norm SVM and ℓ_2-norm SVM, respectively.

An illustrative example for the SVM problem is given in Fig. 4.2. The larger the margin is, the farther the data points are from the hyperplane and so the better the classification is. This is the key point for the SVM approach, to find a hyperplane that maximizes the margin, which is equivalent to minimizing the norm of the weight vector.

The optimization problems (4.4) and (4.5) can be solved by very efficient algorithms [88–91]. For example, if the weight vector is sparse, the complexity of the algorithm in [88] scales linearly in both the number of features (size of the weight vector $\mathbf{w}$) and the number of training samples $|\mathcal{D}|$. For arbitrary weight vectors, the complexity of the algorithms in [89–91] scales linearly in the number of features and super-linearly in the number of training samples. A good review of efficient methods for solving (4.4) and (4.5) can also be found in [92].

In this chapter, we exploit the linear SVM framework described above to develop channel estimation and data detection algorithms for one-bit massive MIMO systems. This idea is motivated by the observation that a one-bit ADC assigns its received signal to one of the two classes $\{+1, -1\}$, which means it is possible to consider the one-bit MIMO system under the SVM framework.

4.3.2 Proposed SVM-based Channel Estimation

Uncorrelated Channels

First, uncorrelated channels are considered. The channel elements are assumed to be i.i.d. as $\mathcal{CN}(0,1)$. In order to estimate the channel, a pilot sequence $\mathbf{X}_{\mathrm{t}}^{\mathbb{C}} \in \mathbb{C}^{U \times T_{\mathrm{t}}}$ of length T_{t} is

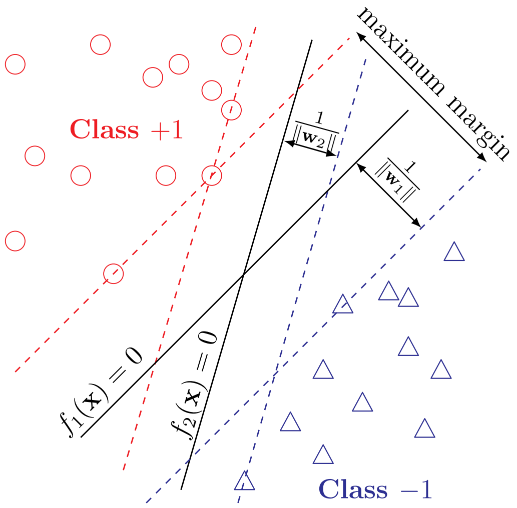


Figure 4.2: An illustrative example of SVM. The hyperplane $f_2(\mathbf{x}) = \mathbf{w}_2^T\mathbf{x} + b_2 = 0$ correctly classifies the data points but its margin is not the largest possible. The hyperplane $f_1(\mathbf{x}) = \mathbf{w}_1^T\mathbf{x} + b_1 = 0$ not only correctly classifies the data points and its margin is also the maximum, thus f_1 is the SVM solution.

used to generate the training data

$$\mathbf{Y}_{\mathrm{t}}^{\mathbb{C}} = \mathrm{sign}\left(\mathbf{H}^{\mathbb{C}}\mathbf{X}_{\mathrm{t}}^{\mathbb{C}} + \mathbf{Z}_{\mathrm{t}}^{\mathbb{C}}\right). \tag{4.6}$$

For convenience in later derivations, we convert the notation in (4.6) to the real domain as

$$\mathbf{Y}_{\mathrm{t}} = \mathrm{sign}\left(\mathbf{H}_{\mathrm{t}}\mathbf{X}_{\mathrm{t}} + \mathbf{Z}_{\mathrm{t}}\right), \tag{4.7}$$

where

$$\mathbf{Y}_{\mathrm{t}} = \left[\Re\{\mathbf{Y}_{\mathrm{t}}^{\mathbb{C}}\}, \Im\{\mathbf{Y}_{\mathrm{t}}^{\mathbb{C}}\}\right] = \left[\mathbf{y}_{\mathrm{t},1}, \mathbf{y}_{\mathrm{t},2}, \ldots, \mathbf{y}_{\mathrm{t},N}\right]^T, \tag{4.8}$$

$$\mathbf{H}_{\mathrm{t}} = \left[\Re\{\mathbf{H}^{\mathbb{C}}\}, \Im\{\mathbf{H}^{\mathbb{C}}\}\right] = \left[\mathbf{h}_{\mathrm{t},1}, \mathbf{h}_{\mathrm{t},2}, \ldots, \mathbf{h}_{\mathrm{t},N}\right]^T, \tag{4.9}$$

$$\mathbf{Z}_{\mathrm{t}} = \left[\Re\{\mathbf{Z}_{\mathrm{t}}^{\mathbb{C}}\}, \Im\{\mathbf{Z}_{\mathrm{t}}^{\mathbb{C}}\}\right] = \left[\mathbf{z}_{\mathrm{t},1}, \mathbf{z}_{\mathrm{t},2}, \ldots, \mathbf{z}_{\mathrm{t},N}\right]^T, \tag{4.10}$$

and

$$\mathbf{X}_{\mathrm{t}} = \begin{bmatrix} \Re\{\mathbf{X}_{\mathrm{t}}^{\mathbb{C}}\} & \Im\{\mathbf{X}_{\mathrm{t}}^{\mathbb{C}}\} \\ -\Im\{\mathbf{X}_{\mathrm{t}}^{\mathbb{C}}\} & \Re\{\mathbf{X}_{\mathrm{t}}^{\mathbb{C}}\} \end{bmatrix} = [\mathbf{x}_{\mathrm{t},1}, \mathbf{x}_{\mathrm{t},2}, \ldots, \mathbf{x}_{\mathrm{t},2T_{\mathrm{t}}}]. \tag{4.11}$$

Note that $\mathbf{y}_{\mathrm{t},i}^T \in \{\pm 1\}^{1\times 2T_{\mathrm{t}}}$, $\mathbf{h}_{\mathrm{t},i}^T \in \mathbb{R}^{1\times 2U}$, and $\mathbf{z}_{\mathrm{t},i}^T \in \mathbb{R}^{1\times 2T_{\mathrm{t}}}$ with $i \in \{1, 2, \ldots, N\}$ represent the i^{th} rows of $\mathbf{Y}_{\mathrm{t}}$, $\mathbf{H}_{\mathrm{t}}$, and $\mathbf{Z}_{\mathrm{t}}$, respectively. However, $\mathbf{x}_{\mathrm{t},n} \in \mathbb{R}^{2U\times 1}$ with $n \in \{1, 2, \ldots, 2T_{\mathrm{t}}\}$ is the n^{th} column of $\mathbf{X}_{\mathrm{t}}$.

It can be seen from (4.9) that estimating $\{\mathbf{h}_{\mathrm{t},i}\}_{i=1,2,\ldots,N}$ is equivalent to estimating $\bar{\mathbf{H}}$. Here, the channel estimation problem is formulated in terms of $\mathbf{h}_{\mathrm{t},i}$. Let

$$\mathbf{y}_{\mathrm{t},i} = [y_{\mathrm{t},i,1}, y_{\mathrm{t},i,2}, \ldots, y_{\mathrm{t},i,2T_{\mathrm{t}}}]^T \text{ and}$$
$$\mathbf{z}_{\mathrm{t},i} = [z_{\mathrm{t},i,1}, z_{\mathrm{t},i,2}, \ldots, z_{\mathrm{t},i,2T_{\mathrm{t}}}]^T,$$

then we have

$$y_{\mathrm{t},i,n} = \operatorname{sign}\left(\mathbf{h}_{\mathrm{t},i}^T \mathbf{x}_{\mathrm{t},n} + z_{\mathrm{t},i,n}\right). \tag{4.12}$$

It is stressed that the estimation of $\mathbf{h}_{\mathrm{t},i}$ in (5.6) can be interpreted as an SVM binary classification problem. More specifically, $\{\mathbf{x}_{\mathrm{t},n}, y_{\mathrm{t},i,n}\}_{n=1,\ldots,2T_{\mathrm{t}}}$ plays the role of the training data set $\mathcal{D}$. The channel $\mathbf{h}_{\mathrm{t},i}$ acts as the weight vector and $z_{\mathrm{t},i,n}$ can be viewed as the bias. Hence, the SVM classification formulation can be exploited to estimate $\mathbf{h}_{\mathrm{t},i}$ by solving the following optimization problem:

$$\begin{aligned} \underset{\{\mathbf{h}_{\mathrm{t},i},\xi_n\}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{h}_{\mathrm{t},i}\|^2 + C\sum_{n=1}^{2T_{\mathrm{t}}} \ell(\xi_n) \\ \text{subject to} \quad & y_{\mathrm{t},i,n}\mathbf{h}_{\mathrm{t},i}^T \mathbf{x}_{\mathrm{t},n} \geq 1 - \xi_n, \\ & \xi_n \geq 0, \quad n = 1, 2, \ldots, 2T_{\mathrm{t}}. \end{aligned} \tag{4.13}$$

Here, the bias is discarded because the $\{z_{\text{t},i,n}\}$ are random noise with zero mean. In addition, at infinite SNR, (5.6) becomes $y_{\text{t},i,n} = \text{sign}\left(\mathbf{h}_{\text{t},i}^T \mathbf{x}_{\text{t},n}\right)$, which has no bias. It should be noted that (4.13) only depends on a single index i, and so its solution is the estimate for the i^{th} row of the channel matrix $\mathbf{H}^{\mathbb{C}}$, i.e., the channel vector from the U users to the ith receive antenna. This means we have N separate optimization problems of the same form (4.13), which is an advantage of the proposed SVM-based method since these N optimization problems can be solved in parallel.

Let $\tilde{\mathbf{h}}_{\text{t},i}$ denote the solution of (4.13). This solution provides an estimate of the channel "direction", but the magnitude of $\tilde{\mathbf{h}}_{\text{t},i}$ is determined by the definition of the SVM margin, which in turn defines the inequality constraints in (4.13). In fact, the instantaneous magnitude of $\mathbf{h}_{\text{t},i}$ is not identifiable [48] since $a\mathbf{h}_{\text{t},i}$ for any $a > 0$ will produce the same data set $\{y_{t,i,n}\}$:

$$y_{\text{t},i,n} = \text{sign}\left(\mathbf{h}_{\text{t},i}^T \mathbf{x}_{\text{t},n}\right) = \text{sign}\left(a\mathbf{h}_{\text{t},i}^T \mathbf{x}_{\text{t},n}\right), \text{ with } a > 0.$$

Since in the considered model we assume that the $2U$ elements of $\mathbf{h}_{\text{t},i}$ are each independent with variance 1/2, the SVM solution is scaled so that the corresponding channel estimate has a squared norm of U:

$$\hat{\mathbf{h}}_{\text{t},i} = \frac{\sqrt{U}\tilde{\mathbf{h}}_{\text{t},i}}{\|\tilde{\mathbf{h}}_{\text{t},i}\|}. \tag{4.14}$$

This rescaling choice is found to provide the best estimation accuracy.

Remark 1: The soft-SVM method in [37] does not maximize the margin, but instead calculates $\mathbf{h}_{\text{t},i}$ such that the condition $y_{\text{t},i,n}\mathbf{h}_{\text{t},i}^T \mathbf{x}_{\text{t},n} > 0$ is satisfied for as many n as possible. However, due to the noise component $z_{\text{t},i,n}$, the condition $y_{\text{t},i,n}\mathbf{h}_{\text{t},i}^T \mathbf{x}_{\text{t},n} > 0$ may not be satisfied even with the true channel vector $\mathbf{h}_{\text{t},i}$. The proposed method exploits the original idea of SVM by maximizing the margin achieved by the linear discriminator. The introduction of the slack variables in the problem circumvents the strict constraint $y_{\text{t},i,n}\mathbf{h}_{\text{t},i}^T \mathbf{x}_{\text{t},n} > 0$.

Remark 2: Without slack variables, the problem in (4.13)

$$\begin{aligned} \underset{\{\mathbf{h}_{\mathrm{t},i}\}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{h}_{\mathrm{t},i}\|^2 \\ \text{subject to} \quad & y_{\mathrm{t},i,n}\mathbf{h}_{\mathrm{t},i}^T\mathbf{x}_{\mathrm{t},n} \geq 1, \quad n = 1, 2, \ldots, 2T_{\mathrm{t}}, \end{aligned} \tag{4.15}$$

is similar to the form in (4.4). For $\mathbf{h}_{\mathrm{t},i} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ we have

$$p(\mathbf{h}_{\mathrm{t},i}) = \frac{1}{\sqrt{(2\pi)^{2U}}} \exp\left\{-\frac{1}{2}\|\mathbf{h}_{\mathrm{t},i}\|^2\right\}, \tag{4.16}$$

and hence the optimization problem in (4.15) can be read as maximizing the pdf of $\mathbf{h}_{\mathrm{t},i}$ subject to the constraints $y_{\mathrm{t},i,n}\mathbf{h}_{\mathrm{t},i}^T\mathbf{x}_{\mathrm{t},n} \geq 1$ for $n = 1, 2, \ldots, 2T_{\mathrm{t}}$. Thus, the SVM approach can be interpreted as finding the channel $\mathbf{h}_{\mathrm{t},i}$ that attains the highest likelihood under the constraints realized by the measured data. This observation will be used next to modify the SVM-based channel estimator when the channel is spatially correlated. Note that the work in [37] only considers uncorrelated channels.

Spatially Correlated Channels

Let $\mathbf{H}^{\mathbb{C}} = [\mathbf{h}_1^{\mathbb{C}}, \ldots, \mathbf{h}_U^{\mathbb{C}}]$, and so $\mathbf{h}_u^{\mathbb{C}} \in \mathbb{C}^{N\times 1}$ is the u^{th} column of $\mathbf{H}^{\mathbb{C}}$. Here, it is assumed that the elements of $\mathbf{h}_u^{\mathbb{C}}$ are correlated, or in other words that the channels associated with different antennas are correlated. Let $\mathbf{h}_u^{\mathbb{C}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}_u^{\mathbb{C}})$ and $\mathbf{h}^{\mathbb{C}} = \mathrm{vec}(\mathbf{H}^{\mathbb{C}})$, then $\mathbf{h}^{\mathbb{C}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}^{\mathbb{C}})$ where $\mathbf{C}^{\mathbb{C}} = \mathrm{blkdiag}(\mathbf{C}_1^{\mathbb{C}}, \mathbf{C}_2^{\mathbb{C}}, \ldots, \mathbf{C}_U^{\mathbb{C}})$. The pdf of $\mathbf{h}^{\mathbb{C}}$ is

$$\begin{aligned} p(\mathbf{h}^{\mathbb{C}}) &= \frac{1}{\pi^{UN}\sqrt{\det(\mathbf{C}^{\mathbb{C}})}} \exp\left\{-(\mathbf{h}^{\mathbb{C}})^H(\mathbf{C}^{\mathbb{C}})^{-1}\mathbf{h}^{\mathbb{C}}\right\} && (4.17) \\ &= \frac{1}{\pi^{UN}\sqrt{\det(\mathbf{C}^{\mathbb{C}})}} \exp\left\{-\sum_{u=1}^{U}(\mathbf{h}_u^{\mathbb{C}})^H(\mathbf{C}_u^{\mathbb{C}})^{-1}\mathbf{h}_u^{\mathbb{C}}\right\}. && (4.18) \end{aligned}$$

The exponent term in (4.17) becomes a sum in (4.18) because $\mathbf{C}^{\mathbb{C}}$ is a block diagonal matrix, whose main-diagonal blocks are $\mathbf{C}_1^{\mathbb{C}}, \mathbf{C}_2^{\mathbb{C}}, \ldots, \mathbf{C}_U^{\mathbb{C}}$. Letting

$$\mathbf{h}_u = \begin{bmatrix} \Re\{\mathbf{h}_u^{\mathbb{C}}\} \\ \Im\{\mathbf{h}_u^{\mathbb{C}}\} \end{bmatrix} \text{ and } \mathbf{C}_u = \begin{bmatrix} \Re\{\mathbf{C}_u^{\mathbb{C}}\} & -\Im\{\mathbf{C}_u^{\mathbb{C}}\} \\ \Im\{\mathbf{C}_u^{\mathbb{C}}\} & \Re\{\mathbf{C}_u^{\mathbb{C}}\} \end{bmatrix},$$

the exponent term in (4.18) can be rewritten as $\sum_{u=1}^{U} \mathbf{h}_u^T \mathbf{C}_u^{-1} \mathbf{h}_u$.

To maximize the likelihood of $\mathbf{h}^{\mathbb{C}}$ subject to the constraints $y_{\mathrm{t},i,n}\mathbf{h}_{\mathrm{t},i}^T\mathbf{x}_{\mathrm{t},n} \geq 1$ with $i = 1, 2, \ldots, N$ and $n = 1, 2, \ldots, 2T_{\mathrm{t}}$, we can follow the intuition in (4.15) to formulate the following optimization problem:

$$\begin{aligned} \underset{\{\mathbf{H}^{\mathbb{C}}\}}{\text{minimize}} \quad & \frac{1}{2}\sum_{u=1}^{U} \|\mathbf{h}_u^T \mathbf{C}_u^{-1} \mathbf{h}_u\|^2 \\ \text{subject to} \quad & y_{\mathrm{t},i,n}\mathbf{h}_{\mathrm{t},i}^T\mathbf{x}_{\mathrm{t},n} \geq 1, \\ & i = 1, 2, \ldots, N \text{ and } n = 1, 2, \ldots, 2T_{\mathrm{t}}. \end{aligned} \tag{4.19}$$

In the above optimization problem, it is important to note that $\mathbf{h}_u \in \mathbb{R}^{2N\times 1}$ represents the u^{th} column of $\mathbf{H}^{\mathbb{C}}$, but $\mathbf{h}_{\mathrm{t},i}^T$ represents the i^{th} row of $\mathbf{H}^{\mathbb{C}}$. This means the objective function of (4.19) depends on the columns of $\mathbf{H}^{\mathbb{C}}$, but the constraints depend on the rows of $\mathbf{H}^{\mathbb{C}}$. Therefore, we cannot decompose (4.19) into smaller independent problems. In other words, the whole channel matrix $\mathbf{H}^{\mathbb{C}}$ has to be jointly estimated.

Note that the margin $\mathbf{h}_u^T \mathbf{C}_u^{-1} \mathbf{h}_u$ in (4.19) is measured using the Mahalanobis distance [97] rather than the Euclidean metric used in the standard SVM approach. The optimization problem in (4.19) can also be generalized by including slack variables as

$$\begin{aligned} \underset{\{\mathbf{H}^{\mathbb{C}},\xi_{i,n}\}}{\text{minimize}} \quad & \frac{1}{2}\sum_{u=1}^{U}\|\mathbf{h}_u^T\mathbf{C}_u^{-1}\mathbf{h}_u\|^2 + C\sum_{i=1}^{N}\sum_{n=1}^{2T_{\mathrm{t}}}\ell(\xi_{i,n}) \\ \text{subject to} \quad & y_{\mathrm{t},i,n}\mathbf{h}_{\mathrm{t},i}^T\mathbf{x}_{\mathrm{t},n} \geq 1-\xi_{i,n} \text{ with } \xi_{i,n}\geq 0, \\ & i=1,2,\ldots,N \text{ and } n=1,2,\ldots,2T_{\mathrm{t}}. \end{aligned} \tag{4.20}$$

Although the form of the objective function in (4.20) is different from that in conventional SVM problems, (4.20) can still be solved efficiently since it is a convex optimization problem. Let $\tilde{\mathbf{H}}$ be the solution of (4.20), then the channel estimate $\hat{\mathbf{H}}$ is defined as

$$\hat{\mathbf{H}} = \frac{\sqrt{UN}\tilde{\mathbf{H}}}{\|\tilde{\mathbf{H}}\|_{\mathrm{F}}},$$

where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm. This normalization step is similar to that for the case of uncorrelated channels, except a different coefficient $\sqrt{UN}$ is used since we jointly estimate the whole channel matrix and $\mathbb{E}[\|\mathbf{H}^{\mathbb{C}}\|_{\mathrm{F}}] = \sqrt{UN}$.

4.3.3 Proposed Two-Stage SVM-based Data Detection

This section proposes a two-stage SVM-based method for data detection with 1-bit ADCs. The data detection is first formulated as an SVM problem. A second stage is then employed to refine the solution from the first stage. Let $\mathbf{X}_{\mathrm{d}}^{\mathbb{C}} = [\mathbf{x}_{\mathrm{d},1}^{\mathbb{C}}, \mathbf{x}_{\mathrm{d},2}^{\mathbb{C}}, \ldots, \mathbf{x}_{\mathrm{d},T_{\mathrm{d}}}^{\mathbb{C}}] \in \mathbb{C}^{U\times T_{\mathrm{d}}}$ be the transmitted data sequence of length T_{d}. The received data signal is given as

$$\mathbf{Y}_{\mathrm{d}}^{\mathbb{C}} = \mathrm{sign}\left(\mathbf{H}^{\mathbb{C}}\mathbf{X}_{\mathrm{d}}^{\mathbb{C}} + \mathbf{Z}_{\mathrm{d}}^{\mathbb{C}}\right). \tag{4.21}$$

The above equation is also converted to the real domain as

$$\mathbf{Y}_{\mathrm{d}} = \operatorname{sign}\left(\mathbf{H}_{\mathrm{d}}\mathbf{X}_{\mathrm{d}} + \mathbf{Z}_{\mathrm{d}}\right) \tag{4.22}$$

where

$$\mathbf{Y}_{\mathrm{d}} = \begin{bmatrix} \Re\{\mathbf{Y}_{\mathrm{d}}^{\mathbb{C}}\} \\ \\ \Im\{\mathbf{Y}_{\mathrm{d}}^{\mathbb{C}}\} \end{bmatrix} = [\mathbf{y}_{\mathrm{d},1}, \mathbf{y}_{\mathrm{d},2}, \ldots, \mathbf{y}_{\mathrm{d},T_{\mathrm{d}}}], \tag{4.23}$$

$$\mathbf{X}_{\mathrm{d}} = \begin{bmatrix} \Re\{\mathbf{X}_{\mathrm{d}}^{\mathbb{C}}\} \\ \\ \Im\{\mathbf{X}_{\mathrm{d}}^{\mathbb{C}}\} \end{bmatrix} = [\mathbf{x}_{\mathrm{d},1}, \mathbf{x}_{\mathrm{d},2}, \ldots, \mathbf{x}_{\mathrm{d},T_{\mathrm{d}}}], \tag{4.24}$$

$$\mathbf{Z}_{\mathrm{d}} = \begin{bmatrix} \Re\{\mathbf{Z}_{\mathrm{d}}^{\mathbb{C}}\} \\ \\ \Im\{\mathbf{Z}_{\mathrm{d}}^{\mathbb{C}}\} \end{bmatrix} = [\mathbf{z}_{\mathrm{d},1}, \mathbf{z}_{\mathrm{d},2}, \ldots, \mathbf{z}_{\mathrm{d},T_{\mathrm{d}}}], \text{ and} \tag{4.25}$$

$$\mathbf{H}_{\mathrm{d}} = \begin{bmatrix} \Re\{\mathbf{H}^{\mathbb{C}}\} & -\Im\{\mathbf{H}^{\mathbb{C}}\} \\ \\ \Im\{\mathbf{H}^{\mathbb{C}}\} & \Re\{\mathbf{H}^{\mathbb{C}}\} \end{bmatrix} = [\mathbf{h}_{\mathrm{d},1}, \mathbf{h}_{\mathrm{d},2}, \ldots, \mathbf{h}_{\mathrm{d},2N}]^T. \tag{4.26}$$

Here, $\mathbf{y}_{\mathrm{d},m} \in \{\pm 1\}^{2N\times 1}$, $\mathbf{x}_{\mathrm{d},m} \in \mathbb{R}^{2U\times 1}$, and $\mathbf{z}_{\mathrm{d},m} \in \mathbb{R}^{2N\times 1}$ with $m \in \{1, 2, \ldots, T_{\mathrm{d}}\}$ are the m^{th} columns of $\mathbf{Y}_{\mathrm{d}}$, $\mathbf{X}_{\mathrm{d}}$, and $\mathbf{Z}_{\mathrm{d}}$, respectively. However, $\mathbf{h}_{\mathrm{d},i'}^T \in \mathbb{R}^{1\times 2U}$ with $i' \in \{1, 2, \ldots, 2N\}$ represents the i'^{th} row of $\mathbf{H}_{\mathrm{d}}$.

It can be noted that the real and imaginary parts in (4.8)–(4.11) are stacked side-by-side, but they are stacked on top of each other in (4.23)–(4.26). This is due to the exchange in the role of the channel and the data matrices. In the formulation for channel estimation in (4.8)–(4.11), each row of the channel matrix is treated as the weight vector and the columns of the pilot data matrix are used as the training data points. On the other hand, the data detection formulation in (4.23)–(4.26) treats each column of the to-be-decoded data matrix as the weight vector and the rows of the channel matrix as the training data points.

It should also be noted that the pilot sequence and the data sequence are assumed to experience the same block-fading channel. Although the two channel matrices $\mathbf{H}_{\mathrm{t}}$ in (4.9) and $\mathbf{H}_{\mathrm{d}}$ in (4.26) are constructed differently, they still depend on the same channel $\mathbf{H}^{\mathbb{C}}$. Let

$$\begin{aligned}\mathbf{y}_{\mathrm{d},m} &= [y_{\mathrm{d},m,1}, y_{\mathrm{d},m,2}, \ldots, y_{\mathrm{d},m,2N}]^T \text{ and}\\ \mathbf{z}_{\mathrm{d},m} &= [z_{\mathrm{d},m,1}, z_{\mathrm{d},m,2}, \ldots, z_{\mathrm{d},m,2N}]^T,\end{aligned}$$

then we have

$$y_{\mathrm{d},m,i'} = \operatorname{sign}\left(\mathbf{h}_{\mathrm{d},i'}^T \mathbf{x}_{\mathrm{d},m} + z_{\mathrm{d},m,i'}\right). \tag{4.27}$$

It is observed that the estimation of $\mathbf{x}_{\mathrm{d},m}$ can also be interpreted as an SVM binary classification problem. More specifically, we can treat $\mathbf{x}_{\mathrm{d},m}$ as the weight vector and the set $\{\hat{\mathbf{h}}_{\mathrm{d},i'}, y_{\mathrm{d},m,i'}\}_{i'=1,\ldots,2N}$ as the training set, where $\hat{\mathbf{h}}_{\mathrm{d},i'}$ is the channel estimate of $\mathbf{h}_{\mathrm{d},i'}$ obtained as explained above. The following optimization problem provides the first-stage in finding $\mathbf{x}_{\mathrm{d},m}$:

$$\begin{aligned}\underset{\{\mathbf{x}_{\mathrm{d},m},\xi_{i'}\}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{x}_{\mathrm{d},m}\|^2 + C\sum_{i=1}^{2N} \ell(\xi_{i'})\\ \text{subject to} \quad & y_{\mathrm{d},m,i'}\mathbf{x}_{\mathrm{d},m}^T \hat{\mathbf{h}}_{\mathrm{d},i'} \geq 1 - \xi_{i'},\\ & \xi_{i'} \geq 0, \quad i' = 1, 2, \ldots, 2N,\end{aligned} \tag{4.28}$$

where the bias is discarded as in the channel estimation problem. Let $\tilde{\mathbf{x}}_{\mathrm{d},m}$ denote the solution of (4.28) and let $\grave{\mathbf{x}}_{\mathrm{d},m}$ be the normalized version of $\tilde{\mathbf{x}}_{\mathrm{d},m}$ as

$$\grave{\mathbf{x}}_{\mathrm{d},m} = \frac{\sqrt{U}\tilde{\mathbf{x}}_{\mathrm{d},m}}{\|\tilde{\mathbf{x}}_{\mathrm{d},m}\|}. \tag{4.29}$$

This normalization step is also used in [31] in order to make the power of the estimated signal equal the power of the transmitted signal.

Let $\grave{\mathbf{x}}_{\mathrm{d},m} = [\grave{x}_{\mathrm{d},m,1}, \ldots, \grave{x}_{\mathrm{d},m,2U}]^T$, and define the first-stage detected data vector $\check{\mathbf{x}}_{\mathrm{d},m} =$

$[\check{x}_{\mathrm{d},m,1}, \ldots, \check{x}_{\mathrm{d},m,U}]^T$ obtained using symbol-by-symbol detection as

$$\check{x}_{\mathrm{d},m,u} = \underset{x \in \mathcal{M}^{\mathbb{C}}}{\arg\min} \left| (\grave{x}_{\mathrm{d},m,u} + j\grave{x}_{\mathrm{d},m,u+U}) - x \right|, \tag{4.30}$$

where $u \in \mathcal{U}$ and $\mathcal{M}^{\mathbb{C}}$ represents the signal constellation (e.g., QPSK or 16-QAM). The solution to (4.30) is referred to as the stage 1 solution. To further improve the detection performance, a simple but efficient second detection stage is proposed as follows.

First, a candidate set $\mathcal{X}_u$ for each $x^{\mathbb{C}}_{\mathrm{d},m,u}$ is created using $\check{x}_{\mathrm{d},m,u}$ and $\grave{x}_{\mathrm{d},m,u} + j\grave{x}_{\mathrm{d},m,u+U}$ as

$$\mathcal{X}_u = \left\{ \acute{x} \in \mathcal{M}^{\mathbb{C}} \middle| \frac{\left| (\grave{x}_{\mathrm{d},m,u} + j\grave{x}_{\mathrm{d},m,u+U}) - \acute{x} \right|}{\left| (\grave{x}_{\mathrm{d},m,u} + j\grave{x}_{\mathrm{d},m,u+U}) - \check{x}_{\mathrm{d},m,u} \right|} < \nu \right\} \tag{4.31}$$

where $\nu \geq 1$ is a parameter that controls the size of $\mathcal{X}_u$. Then the candidate set $\mathcal{X}_{\mathrm{d},m}$ for $\mathbf{x}_{\mathrm{d},m}$ is obtained as

$$\mathcal{X}_{\mathrm{d},m} = \left\{ [\acute{x}_1, \acute{x}_2, \ldots, \acute{x}_U]^T \mid \acute{x}_u \in \mathcal{X}_u, \forall u \in \mathcal{U} \right\}. \tag{4.32}$$

The above candidate set formation was introduced in [31]. However, the detected data vector in [31] is obtained by searching over $\mathcal{X}_{\mathrm{d},m}$ using the ML criterion, and the resulting performance is susceptible to imperfect CSI at high SNRs. This susceptibility has been reported via numerical results in [65], but no justification was given. An explanation for this issue is provided in Appendix D. To deal with the issue, here a different criterion referred to as *minimum weighted Hamming distance* [61] is adopted. Suppose that $\mathcal{X}_{\mathrm{d},m} = \{\acute{\mathbf{x}}_1, \acute{\mathbf{x}}_2, \ldots, \acute{\mathbf{x}}_{|\mathcal{X}_{\mathrm{d},m}|}\}$ and let $\dot{\mathbf{x}}_l = [\Re\{\acute{\mathbf{x}}_l\}^T, \Im\{\acute{\mathbf{x}}_l\}^T]^T$ with $l \in \{1, 2, \ldots, |\mathcal{X}_{\mathrm{d},m}|\}$. The second-stage detected data vector $\hat{\mathbf{x}}_{\mathrm{d},m}$ is defined as $\hat{\mathbf{x}}_{\mathrm{d},m} = \acute{\mathbf{x}}_{\hat{l}}$ where

$$\hat{l} = \underset{l \in \{1, \ldots, |\mathcal{X}_{\mathrm{d},m}|\}}{\arg\min} \; d_{\mathrm{w}} \left(\mathbf{y}_{\mathrm{d},m}, \mathrm{sign}(\hat{\mathbf{H}}_{\mathrm{d}} \dot{\mathbf{x}}_l) \right). \tag{4.33}$$

Here, $\hat{\mathbf{H}}_\mathrm{d}$ is the channel estimate of $\mathbf{H}_\mathrm{d}$ and $d_\mathrm{w}(\cdot,\cdot)$ is the weighted Hamming distance defined in [61].

The minimum weighted Hamming distance criterion above was shown to be statistically efficient [61]. However, the OSD method proposed in [61] requires a preprocessing stage whose computational complexity is proportional to $2^{N_\mathrm{s}}|\mathcal{M}^\mathbb{C}|^U$ for each channel realization. Here $N_\mathrm{s} = 2N/G$ where $G \geq 1$ is an integer. The exponential computational complexity of OSD is a significant drawback in large-scale system implementation. The proposed SVM-based data detection method in this paper can address this complexity issue since the optimization problem (4.28) can be solved by very efficient algorithms [88, 92, 98].

4.3.4 Proposed SVM-based Joint CE-DD

In 1-bit ADC systems, the channel estimation accuracy can be improved by increasing the length of the pilot training sequence, but not necessarily by increasing the SNR [33]. For this reason, an SVM-based joint CE-DD method is here proposed to effectively improve the channel estimate without lengthening the pilot training sequence. The idea is to use the detected data vectors from the two-stage SVM-based method together with the pilot data vectors to obtain a refined channel estimate and then use this refined channel estimate to improve the data detection performance.

Let $\hat{\mathbf{X}}_\mathrm{d}^\mathbb{C}$ be the detected version of $\mathbf{X}_\mathrm{d}^\mathbb{C}$ using the proposed two-stage data detection method and let

$$\hat{\mathbf{X}}_\mathrm{d2} = \begin{bmatrix} \Re\{\hat{\mathbf{X}}_\mathrm{d}^\mathbb{C}\} & \Im\{\hat{\mathbf{X}}_\mathrm{d}^\mathbb{C}\} \\ -\Im\{\hat{\mathbf{X}}_\mathrm{d}^\mathbb{C}\} & \Re\{\hat{\mathbf{X}}_\mathrm{d}^\mathbb{C}\} \end{bmatrix} = [\hat{\mathbf{x}}_{\mathrm{d}2,1}, \ldots, \hat{\mathbf{x}}_{\mathrm{d}2,2T_\mathrm{d}}], \tag{4.34}$$

$$\mathbf{Y}_\mathrm{d2} = \left[\Re\{\mathbf{Y}_\mathrm{d}^\mathbb{C}\}, \Im\{\mathbf{Y}_\mathrm{d}^\mathbb{C}\}\right] = [\mathbf{y}_{\mathrm{d}2,1}, \ldots, \mathbf{y}_{\mathrm{d}2,N}]^T, \tag{4.35}$$

where $\mathbf{y}_{\mathrm{d2},i} = [y_{\mathrm{d2},i,1}, y_{\mathrm{d2},i,2}, \ldots, y_{\mathrm{d2},i,2T_\mathrm{d}}]^T$, $i = 1, \ldots, N$. The channel estimate can be refined by solving the following optimization problem:

$$\begin{aligned}
\underset{\{\mathbf{h}_{\mathrm{t},i}, \xi_{\mathrm{t},n}, \xi_{\mathrm{d},m}\}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{h}_{\mathrm{t},i}\|^2 + C\left(\sum_{n=1}^{2T_\mathrm{t}} \ell(\xi_{\mathrm{t},n}) + \sum_{m=1}^{2T_\mathrm{d}} \ell(\xi_{\mathrm{d},m})\right) \\
\text{subject to} \quad & y_{\mathrm{t},i,n}\mathbf{h}_{\mathrm{t},i}^T \mathbf{x}_{\mathrm{t},n} \geq 1 - \xi_{\mathrm{t},n}, \\
& y_{\mathrm{d2},i,m}\mathbf{h}_{\mathrm{t},i}^T \hat{\mathbf{x}}_{\mathrm{d2},m} \geq 1 - \xi_{\mathrm{d},m}, \\
& \xi_{\mathrm{t},n} \geq 0, \quad n = 1, 2, \ldots, 2T_\mathrm{t}, \\
& \xi_{\mathrm{d},m} \geq 0, \quad m = 1, 2, \ldots, 2T_\mathrm{d}.
\end{aligned} \tag{4.36}$$

In the optimization problem above, we use two sets of slack variables $\{\xi_{\mathrm{t},n}\}$ and $\{\xi_{\mathrm{d},m}\}$, which correspond to the pilot sequence and the data sequence, respectively. This is just for notational convenience, as the two sets of slack variables play the same role. The refined channel estimate obtained by solving (4.36) can now be used for data detection again in (4.28) and (4.33). Note that the channel estimate obtained by (4.13) can be used as the initial solution to (4.36) so that the algorithm will more quickly converge to the optimal solution. Similarly, $\hat{\mathbf{X}}_\mathrm{d}^\mathbb{C}$ can also be used as the initial solution when solving (4.28) with the refined channel estimate. Numerical results in Section 4.5 show that this strategy will hit a certain performance bound as T_d increases.

4.4 Extension to OFDM systems with Frequency-Selective Fading Channels

This section develops SVM-based channel estimation and SVM-based data detection for OFDM systems with frequency-selective fading channels. Consider an uplink multiuser OFDM system with N_c subcarriers. Denote $\mathbf{x}_u^{\mathbb{C},\mathrm{FD}} \in \mathbb{C}^{N_\mathrm{c}\times 1}$ as the OFDM symbol from the u^{th} user in the frequency domain. Throughout the paper, we use the superscripts "TD" and "FD" to refer to Time Domain and Frequency Domain, respectively. A cyclic prefix (CP) of length N_{cp} is added and the number of channel taps $L_{\texttt{tap}}$ is assumed to satisfy $L_{\texttt{tap}} - 1 \leq N_{\text{cp}} \leq N_{\text{c}}$. It is assumed that $L_{\texttt{tap}}$ is known. After removing the CP, the quantized received signal at the i^{th} antenna in the time domain is given by

$$\mathbf{y}_i^{\mathbb{C},\text{TD}} = \text{sign}\left(\sum_{u=1}^{U} \mathbf{G}_{i,u}^{\mathbb{C},\text{TD}} \mathbf{\Gamma}^H \mathbf{x}_u^{\mathbb{C},\text{FD}} + \mathbf{z}_i^{\mathbb{C},\text{TD}}\right) \tag{4.37}$$

where $\mathbf{\Gamma}$ is the DFT matrix of size $N_{\text{c}} \times N_{\text{c}}$; $\mathbf{G}_{i,u}^{\mathbb{C},\text{TD}}$ is a circulant matrix whose first column is $\mathbf{g}_{i,u}^{\mathbb{C},\text{TD}} = [(\mathbf{h}_{i,u}^{\mathbb{C},\text{TD}})^T, 0, \ldots, 0]^T$; and $\mathbf{h}_{i,u}^{\mathbb{C},\text{TD}}$ is the channel vector of the u^{th} user containing the $L_{\texttt{tap}}$ channel taps, which are assumed to be i.i.d. and distributed as $\mathcal{CN}(0, \frac{1}{L_{\texttt{tap}}})$. We also assume block-fading channels where the first OFDM symbol is used for channel estimation and the other OFDM symbols in the block-fading interval are for data transmission. Thus, the problem of channel estimation and data detection are studied separately.

4.4.1 Proposed SVM-based Channel Estimation in OFDM Systems with Frequency-Selective Fading Channels

Denote $\boldsymbol{\phi}_u^{\mathbb{C},\text{TD}} = \mathbf{\Gamma}^H \mathbf{x}_u^{\mathbb{C},\text{FD}}$ and the training matrix $\mathbf{\Phi}_u^{\mathbb{C},\text{TD}}$ as a circulant matrix with first column equal to $\boldsymbol{\phi}_u^{\mathbb{C},\text{TD}}$. The system model in (4.37) can be reorganized as follows:

$$\begin{aligned}
\mathbf{y}_i^{\mathbb{C},\text{TD}} &= \text{sign}\left(\sum_{u=1}^{U} \mathbf{\Phi}_u^{\mathbb{C},\text{TD}} \mathbf{g}_{i,u}^{\mathbb{C},\text{TD}} + \mathbf{z}_i^{\mathbb{C},\text{TD}}\right) \\
&= \text{sign}\left(\sum_{u=1}^{U} \mathbf{\Phi}_{u,L_{\texttt{tap}}}^{\mathbb{C},\text{TD}} \mathbf{h}_{i,u}^{\mathbb{C},\text{TD}} + \mathbf{z}_i^{\mathbb{C},\text{TD}}\right) \\
&= \text{sign}\left(\mathbf{\Phi}_{L_{\texttt{tap}}}^{\mathbb{C},\text{TD}} \mathbf{h}_i^{\mathbb{C},\text{TD}} + \mathbf{z}_i^{\mathbb{C},\text{TD}}\right)
\end{aligned} \tag{4.38}$$

where $\mathbf{\Phi}_{u,L_{\texttt{tap}}}^{\mathbb{C},\mathrm{TD}}$ is the matrix corresponding to the first $L_{\texttt{tap}}$ columns of $\mathbf{\Phi}_{u}^{\mathbb{C},\mathrm{TD}}$, $\mathbf{\Phi}_{L_{\texttt{tap}}}^{\mathbb{C},\mathrm{TD}} = [\mathbf{\Phi}_{1,L_{\texttt{tap}}}^{\mathbb{C},\mathrm{TD}}, \dots, \mathbf{\Phi}_{U,L_{\texttt{tap}}}^{\mathbb{C},\mathrm{TD}}]$, and $\mathbf{h}_{i}^{\mathbb{C},\mathrm{TD}} = [(\mathbf{h}_{i,1}^{\mathbb{C},\mathrm{TD}})^T, \dots, (\mathbf{h}_{i,U}^{\mathbb{C},\mathrm{TD}})^T]^T$.

We also convert (4.38) into the real domain as

$$\mathbf{y}_i^{\mathrm{TD}} = \mathrm{sign}\left(\mathbf{\Phi}_L^{\mathrm{TD}}\mathbf{h}_i^{\mathrm{TD}} + \mathbf{z}_i^{\mathrm{TD}}\right) \tag{4.39}$$

where

$$\begin{aligned}
\mathbf{y}_i^{\mathrm{TD}} &= \left[\Re\{\mathbf{y}_i^{\mathbb{C},\mathrm{TD}}\}^T, \Im\{\mathbf{y}_i^{\mathbb{C},\mathrm{TD}}\}^T\right]^T, \\
\mathbf{h}_i^{\mathrm{TD}} &= \left[\Re\{\mathbf{h}_i^{\mathbb{C},\mathrm{TD}}\}^T, \Im\{\mathbf{h}_i^{\mathbb{C},\mathrm{TD}}\}^T\right]^T, \\
\mathbf{z}_i^{\mathrm{TD}} &= \left[\Re\{\mathbf{z}_i^{\mathbb{C},\mathrm{TD}}\}^T, \Im\{\mathbf{z}_i^{\mathbb{C},\mathrm{TD}}\}^T\right]^T, \text{ and} \\
\mathbf{\Phi}_{L_{\mathrm{tap}}}^{\mathrm{TD}} &= \begin{bmatrix} \Re\{\mathbf{\Phi}_{L_{\mathrm{tap}}}^{\mathbb{C},\mathrm{TD}}\} & -\Im\{\mathbf{\Phi}_{L_{\mathrm{tap}}}^{\mathbb{C},\mathrm{TD}}\} \\ \Im\{\mathbf{\Phi}_{L_{\mathrm{tap}}}^{\mathbb{C},\mathrm{TD}}\} & \Re\{\mathbf{\Phi}_{L_{\mathrm{tap}}}^{\mathbb{C},\mathrm{TD}}\} \end{bmatrix}.
\end{aligned}$$

Denote $\mathbf{y}_i^{\mathrm{TD}} = [y_{i,1}^{\mathrm{TD}}, y_{i,2}^{\mathrm{TD}}, \dots, y_{i,2N_{\mathrm{c}}}^{\mathrm{TD}}]^T$ and $\mathbf{\Phi}_{L_{\mathrm{tap}}}^{\mathrm{TD}} = \left[(\boldsymbol{\phi}_1^{\mathrm{TD}})^T, (\boldsymbol{\phi}_2^{\mathrm{TD}})^T, \dots, (\boldsymbol{\phi}_{2N_{\mathrm{c}}}^{\mathrm{TD}})^T\right]^T$, leading to the following SVM problem for estimating the OFDM channel using one-bit ADCs:

$$\begin{aligned}
&\underset{\{\mathbf{h}_i^{\mathrm{TD}},\xi_n\}}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{h}_i^{\mathrm{TD}}\|^2 + C\sum_{n=1}^{2N_{\mathrm{c}}} \\
&ell(xi_n) \\
&\text{subject to} \quad y_{i,n}^{\mathrm{TD}}\left(\mathbf{h}_i^{\mathrm{TD}}\right)^T\boldsymbol{\phi}_n^{\mathrm{TD}} \geq 1 - \xi_n, \\
&\qquad\qquad\quad \xi_n \geq 0, \quad n = 1, 2, \dots, 2N_{\mathrm{c}}.
\end{aligned} \tag{4.40}$$

Denoting $\tilde{\mathbf{h}}_i^{\mathrm{TD}}$ as the solution of (4.40), then $\mathbf{h}_i^{\mathrm{TD}}$ can be estimated as

$$\hat{\mathbf{h}}_i^{\mathrm{TD}} = \frac{\sqrt{U}\tilde{\mathbf{h}}_i^{\mathrm{TD}}}{\|\tilde{\mathbf{h}}_i^{\mathrm{TD}}\|}. \tag{4.41}$$

Frequency-selective channel estimation methods using one-bit ADCs have been previously proposed in [29, 33], and [49] based on the Bussgang decomposition, additive quantization noise model, and deep learning, respectively. The deep learning method in [49] was shown to outperform the methods of [29, 49] at low SNRs, but its performance tends to degrade as the SNR increases. In addition, the method in [49] requires a training sequence that contains many OFDM symbols, which are required to be orthogonal between different users. In the proposed method, only one OFDM symbol is used in the training phase and all users send their training symbols concurrently.

4.4.2 Proposed SVM-based Data Detection in OFDM Systems with Frequency-Selective Fading Channels

This section describes how SVM can also be used for data detection in OFDM systems with frequency-selective fading channels. The received quantized vector in (4.37) can be rewritten as

$$\mathbf{y}_i^{\mathbb{C},\mathrm{TD}} = \mathrm{sign}\left(\mathbf{G}_i^{\mathbb{C},\mathrm{FD}}\mathbf{x}^{\mathbb{C},\mathrm{FD}} + \mathbf{z}_i^{\mathbb{C},\mathrm{TD}}\right) \tag{4.42}$$

where $\mathbf{G}_i^{\mathbb{C},\mathrm{FD}} = [\mathbf{G}_{i,1}^{\mathbb{C},\mathrm{TD}}\mathbf{\Gamma}^H, \ldots, \mathbf{G}_{i,U}^{\mathbb{C},\mathrm{TD}}\mathbf{\Gamma}^H] \in \mathbb{C}^{N_\mathrm{c}\times N_\mathrm{c}U}$ and $\mathbf{x}^{\mathbb{C},\mathrm{FD}} = [(\mathbf{x}_1^{\mathbb{C},\mathrm{FD}})^T, \ldots, (\mathbf{x}_U^{\mathbb{C},\mathrm{FD}})^T]^T$ is the transmitted symbol vector from the U users over N_c subcarriers. By stacking all the received signal vectors $\left\{\mathbf{y}_i^{\mathbb{C},\mathrm{TD}}\right\}_{i=1,\ldots,N}$ in a column vector, we have the following equation:

$$\mathbf{y}^{\mathbb{C},\mathrm{TD}} = \mathrm{sign}\left(\mathbf{G}^{\mathbb{C},\mathrm{FD}}\mathbf{x}^{\mathbb{C},\mathrm{FD}} + \mathbf{z}^{\mathbb{C},\mathrm{TD}}\right) \tag{4.43}$$

where

$$\mathbf{y}^{\mathbb{C},\mathrm{TD}} = \left[(\mathbf{y}_1^{\mathbb{C},\mathrm{TD}})^T, (\mathbf{y}_2^{\mathbb{C},\mathrm{TD}})^T, \ldots, (\mathbf{y}_N^{\mathbb{C},\mathrm{TD}})^T\right]^T$$

and

$$\mathbf{G}^{\mathbb{C},\mathrm{FD}} = \left[(\mathbf{G}_1^{\mathbb{C},\mathrm{FD}})^T, (\mathbf{G}_2^{\mathbb{C},\mathrm{FD}})^T, \ldots, (\mathbf{G}_N^{\mathbb{C},\mathrm{FD}})^T\right]^T.$$

Let $\mathbf{y}^{\mathrm{TD}}$, $\mathbf{G}^{\mathrm{FD}}$, and $\mathbf{x}^{\mathrm{FD}}$ be the real-valued versions of $\mathbf{y}^{\mathbb{C},\mathrm{TD}}$, $\mathbf{G}^{\mathbb{C},\mathrm{FD}}$, and $\mathbf{x}^{\mathbb{C},\mathrm{FD}}$, respectively. Converting (4.43) to the real domain as in (4.23)–(4.26), we can formulate an SVM problem by treating the rows of $\mathbf{G}^{\mathrm{FD}}$ as the feature vectors, the elements of $\mathbf{y}^{\mathrm{TD}}$ as the binary indicators and $\mathbf{x}^{\mathrm{FD}}$ as the weight vector. The solution of the SVM problem then provides the detected data.

4.5 Numerical Results

This section presents numerical results to show the superiority of the proposed methods against existing ones. For the simulations we set $C = 1$ and parameter γ for the second stage of the SVM-based detection method as $\gamma = \min\left\{\frac{\rho}{10} + 1.5, 3\right\}$ for QPSK and $\gamma = \min\left\{\frac{\rho}{10} + 1.3, 1.5\right\}$ for 16-QAM where ρ is the SNR. These values of γ are chosen empirically to make sure that $|\mathcal{X}|$ is not too large, but still large enough for $\mathcal{X}$ to have a high chance of containing the true transmitted signal vector. The length of the block-fading interval is assumed to be 500 (i.e., $T_{\mathrm{t}} + T_{\mathrm{d}} = 500$) unless otherwise stated. Such an assumption is not stringent for the frequency ranges (e.g., FR1 and FR2) used in 5G systems even with high user mobility, since the high Doppler will be offset by increases in bandwidth and sampling rate.

It should also be noted that the channels considered in all figures of this section are i.i.d uncorrelated, except Fig. 4.6. For flat-fading channel estimation, the k^{th} row of the training matrix $\mathbf{X}_{\mathrm{t}}$ is the $(k+1)^{\mathrm{th}}$ column of the discrete Fourier transform (DFT) matrix of size $T_{\mathrm{t}} \times T_{\mathrm{t}}$. For frequency-selective fading channel estimation, we use orthogonal pilot sequences similar to those in [29, Eq. (23)]. Results in this section are obtained using the ℓ_2-norm SVM formulation as we have found that it provides better performance compared to the ℓ_1-norm formulation. For solving the proposed SVM-based channel estimation and data detection problems, we use the Scikit-learn library [99].

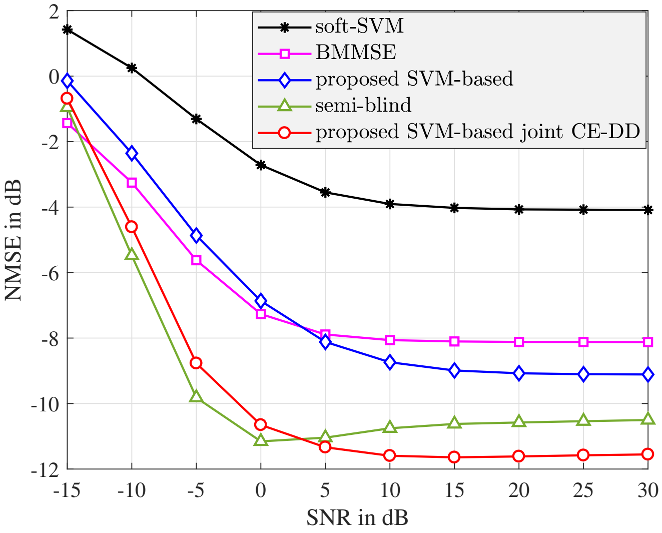


Figure 4.3: NMSE comparison between different channel estimators with $U = 4$, $N = 32$, and $T_{\mathrm{t}} = 20$.

Figure 4.3 presents a performance comparison of different channel estimation methods in terms of NMSE, defined here as $\mathrm{NMSE} = \mathbb{E}\big[\|\hat{\mathbf{H}}^{\mathbb{C}} - \mathbf{H}^{\mathbb{C}}\|_{\mathrm{F}}^{2}\big]/(UN)$, where $\hat{\mathbf{H}}^{\mathbb{C}}$ is an estimate of the channel $\mathbf{H}^{\mathbb{C}}$. It can first be seen that the soft-SVM method performs worse than the other methods. The error floor of the proposed SVM-based channel estimator is lower than that of the BMMSE estimator in [33] and the error floor of the proposed SVM-based joint CE-DD method is also lower than that of the semi-blind channel estimator in [42]. It should be noted that the semi-blind channel estimator is an extension of the BMMSE estimator when the training data set is augmented with some initially detected data vectors. The channel estimators in [33] and [42] perform well at low SNRs. However, they are outperformed by the proposed SVM-based channel estimators at higher SNRs because they use the Bussgang decomposition to obtain a linearized system model that assumes Gaussian inputs to the one-bit quantizers, an assumption that is accurate at low SNRs but less likely to be accurate as the SNR increases. The computational complexity order of the channel estimators studied in these examples is given in Table 4.1.

Figure 4.4 compares the NMSE of BMMSE with the NMSE of the proposed SVM-based method for different values of T_{t}. It is observed that the high-SNR error floor of the BMMSE

Table 4.1: Computational complexity comparison of various channel estimators where N_{iter} is the number of iterations and $f_{\mathrm{sl}}(\cdot)$ is a super-linear function.

Method	**Complexity**
Soft-SVM [37]	$\mathcal{O}(UNT_{\mathrm{t}}N_{\mathrm{iter}})$
BMMSE [33]	$\mathcal{O}(UN^2T_{\mathrm{t}})$
SVM-based	$\mathcal{O}\big(UNT_{\mathrm{t}}f_{\mathrm{sl}}(T_{\mathrm{t}})\big)$
Semi-blind [42]	$\mathcal{O}(UN^2T_{\mathrm{b}}N_{\mathrm{iter}})$
SVM-based joint CE-DD	$\mathcal{O}\big(UNT_{\mathrm{b}}f_{\mathrm{sl}}(T_{\mathrm{b}})\big)$

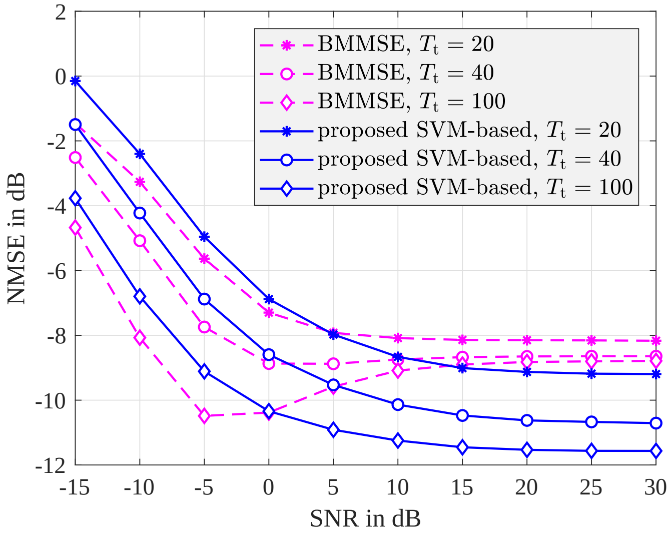


Figure 4.4: NMSE comparison between BMMSE and the proposed SVM-based channel estimator with $U = 4$, $N = 32$, and $T_{\mathrm{t}} \in \{20, 40, 100\}$.

method quickly reaches a bound as T_{t} increases. However, the performance of the proposed SVM-based method improves as T_{t} increases. The error floor of BMMSE even with $T_{\mathrm{t}} = 100$ is still higher than that of the proposed SVM-based method with a much shorter training sequence ($T_{\mathrm{t}} = 20$). The results in Figure 4.4 show that increasing T_{t} can help improve the channel estimation accuracy. However, the spectral efficiency of the system is adversely affected as a result. Thus, the proposed SVM-based joint CE-DD method can help improve both the channel estimation performance and the spectral efficiency.

The effect of T_{d} on the NMSE of the proposed SVM-based joint CE-DD method is studied

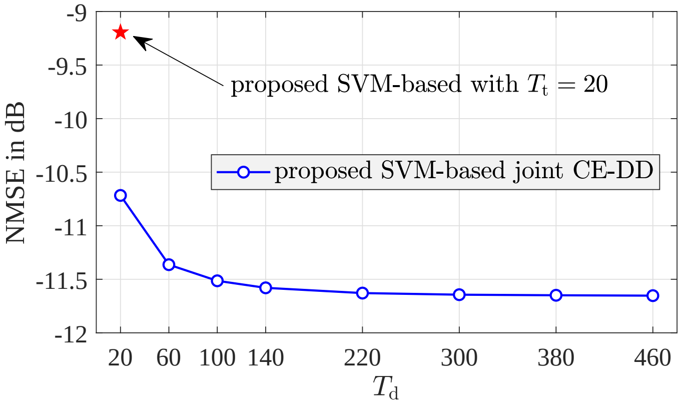


Figure 4.5: Effect of T_d on the NMSE of the proposed SVM-based joint CE-DD with $U = 4$, $N = 32$, and $T_\mathrm{t} = 20$ at $\rho = 30$ dB.

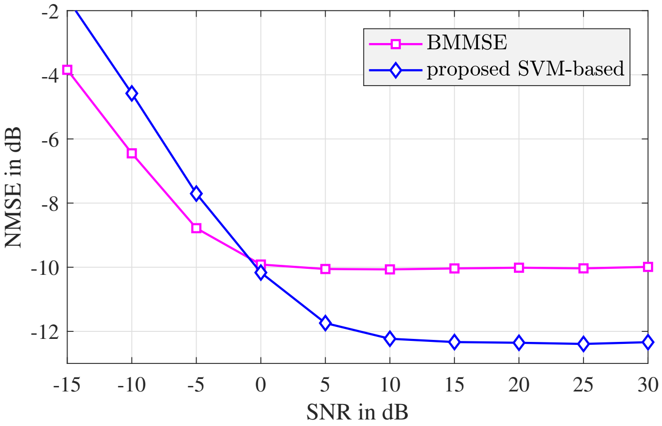


Figure 4.6: NMSE comparison between the BMMSE channel estimator and the proposed SVM-based channel estimator for spatially correlated channels with $U = 4$, $N = 32$, and $T_\mathrm{t} = 20$.

in Figure 4.5. It can be seen that as T_d increases, the channel estimation performance of the SVM-based joint CE-DD method reaches a bound. It is also seen that with a data segment of only about 150 time slots, the channel estimation accuracy can asymptotically reach the bound, which is much better than the performance of using only the training sequence (the red star symbol).

Figure 4.6 presents channel estimation results for spatially correlated channels. We use the same typical urban channel model as in [33]. The power angle spectrum of the channel model follows a Laplacian distribution with an angle spread of $10°$. The simulation results indicate the performance advantage of the proposed SVM-based solution over the BMMSE method

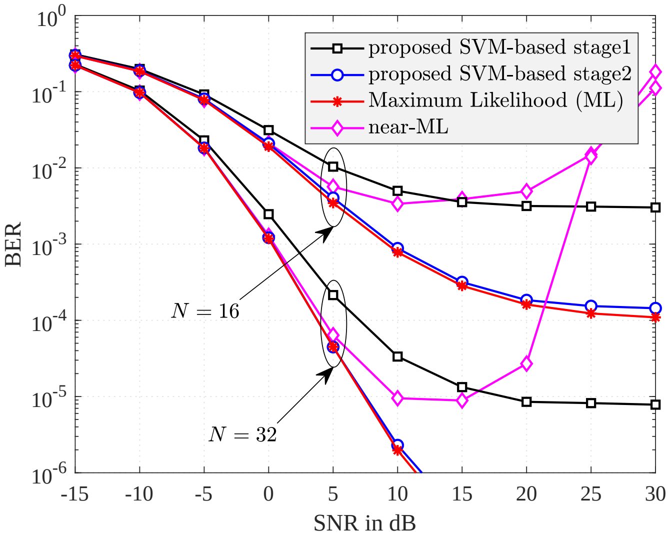


Figure 4.7: Performance comparison between the proposed two-stage SVM-based data detection method and ML detection with perfect CSI, QPSK modulation, and $U = 4$. The average cardinalities of $\mathcal{X}$ for $N = 16$ and $N = 32$ are 2.9352 and 1.6140, respectively.

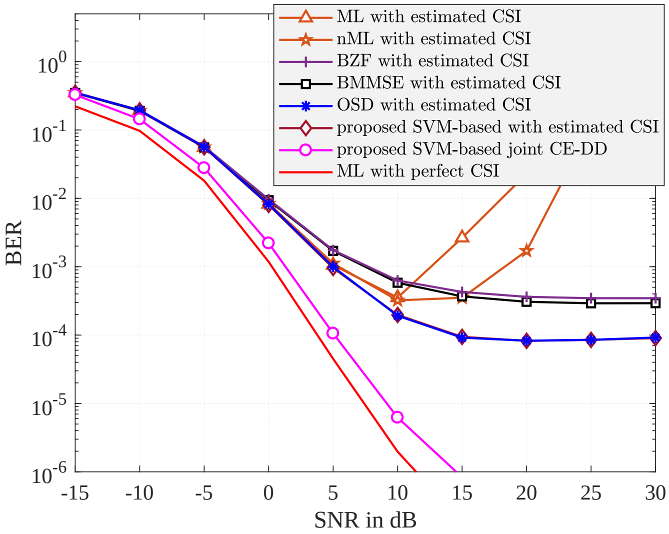


Figure 4.8: Performance comparison between two proposed data detection methods and other existing methods with estimated CSI, QPSK modulation, $N = 32$, $U = 4$, and $T_{\mathrm{t}} = 20$.

at high SNR, and thus justify the SVM-based problem formulation in (4.20).

In Figure 4.7, the proposed two-stage SVM-based data detection method is compared with the ML and nML detection methods for the case of perfect CSI. It is observed that the performance of the proposed method is very close to that of the ML method after two stages. It should be noted that the ML method performs well but it is an exhaustive-search method and so its computational complexity is prohibitively high for large-scale systems. While the nML method is applicable for large-scale systems, it is not robust at high SNRs. This non-robustness occurs regardless of the quality of the CSI, since nML depends on the gradient of a fractional form whose numerator and denominator both rapidly approach zero. It should also be noted that the average cardinalities of $\mathcal{X}$ for $N = 16$ and $N = 32$ are 2.9352 and 1.6140, respectively. This means the second stage of the proposed method is relatively simple to implement since it only has to search over a few candidates.

For the case of imperfect CSI, a bit-error-rate (BER) comparison is provided in Figure 4.8, where the estimated CSI is obtained by the SVM-based channel estimator. Here, the SVM-based joint CE-DD method can be compared with other methods because it also starts with CSI estimated by the SVM-based channel estimator. It is seen that both the ML and nML detection methods are non-robust at high SNRs with imperfect CSI. The susceptibility of ML was also reported in [65]. An explanation for the susceptibility of ML detection can be found in Appendix D. It is also observed that the proposed SVM-based and OSD detection methods give the same performance. However, the complexity order of the proposed SVM-based method is much lower than that of the OSD method as can be seen in Table 4.2. Note that the OSD method requires the choice of two parameters N_{s} and L. Here, we set $N_{\text{s}} = 8$ and $L = 8$ since this choice provides the best performance. The proposed SVM-based joint CE-DD algorithm significantly outperforms other methods and its performance is quite close to the performance of the ML method with perfect CSI. This performance enhancement is due to the refined channel estimate obtained by solving (4.36).

Although the SVM-based and OSD methods give the same performance, the computational complexity of the SVM-based approach is much lower than that of OSD. This is illustrated in Figure 4.9. The average run time required to perform data detection over a block-fading interval of 500 slots is calculated. Note that the OSD method contains two stages: a prepro-

Table 4.2: Computational complexity comparison of data detection methods where $GN_{\mathrm{s}} = 2N$.

Method	**Preprocessing**	**Detection Stage**
BZF [64]	$\mathcal{O}(U^2N)$	$\mathcal{O}(UNT_{\mathrm{d}})$
BMMSE [64]	$\mathcal{O}(N^3)$	
OSD [61]	$\mathcal{O}(2^{N_{\mathrm{s}}}UN\lvert\mathcal{M}\rvert^U)$	$\mathcal{O}\big(UNGLT_{\mathrm{d}}\big)$
ML [31]	$\mathcal{O}(UN\lvert\mathcal{M}\rvert^U)$	$\mathcal{O}\big(N\lvert\mathcal{M}\rvert^U T_{\mathrm{d}}\big)$
nML [31]	–	$\mathcal{O}\big(UNN_{\mathrm{iter}}T_{\mathrm{d}}\big)$
SVM-based	–	$\mathcal{O}\big(UNf_{\mathrm{sl}}(N)T_{\mathrm{d}}\big)$
SVM-based joint CE-DD	–	$\mathcal{O}\big(UNf_{\mathrm{sl}}(T_{\mathrm{b}})T_{\mathrm{b}}\big)$

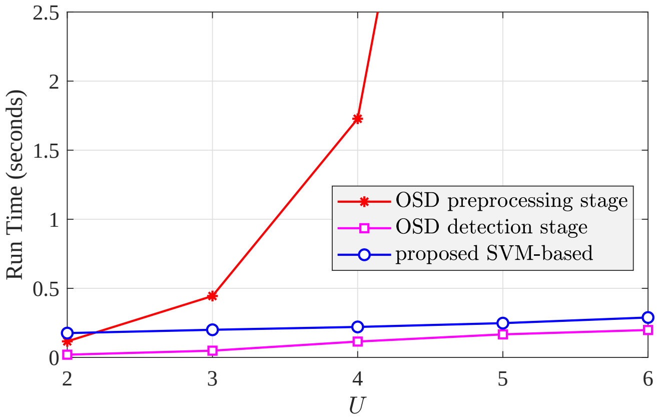


Figure 4.9: Run time comparison between OSD and the proposed SVM-based detection method with QPSK modulation, $N = 32$, and U varies.

cessing stage and a detection stage. It is observed that the OSD method has a low-complexity detection stage. Interestingly, Figure 4.9 indicates that the run time of proposed SVM-based method is comparable to that of the OSD detection stage. However, the OSD method requires a high-complexity preprocessing stage, which scales exponentially with the number of users. This makes the total complexity of the OSD method much higher than that of the SVM-based method, as observed in the figure.

Figure 4.10 and Figure 4.11 provide BER comparisons between the proposed SVM-based data detection methods and other existing methods with QPSK and 16-QAM modulations using the CSI estimated by the SVM-based channel estimator. Due to their high computational

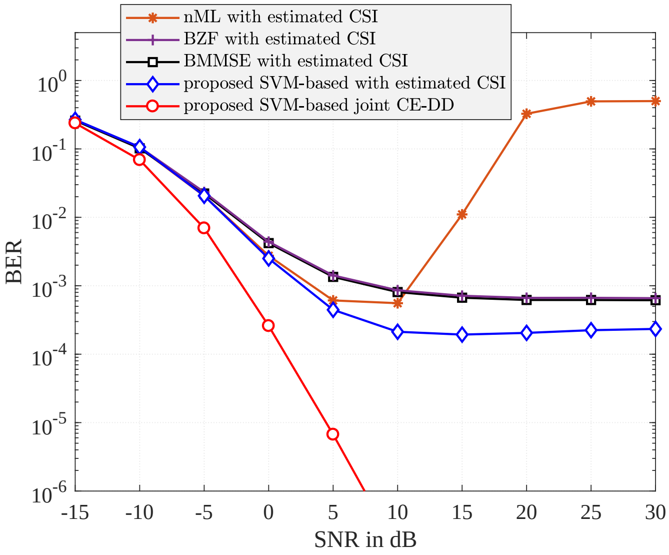


Figure 4.10: Performance comparison between two proposed data detection methods and other existing methods with estimated CSI, QPSK modulation, $N = 64$, $U = 8$, and $T_{\mathrm{t}} = 40$.

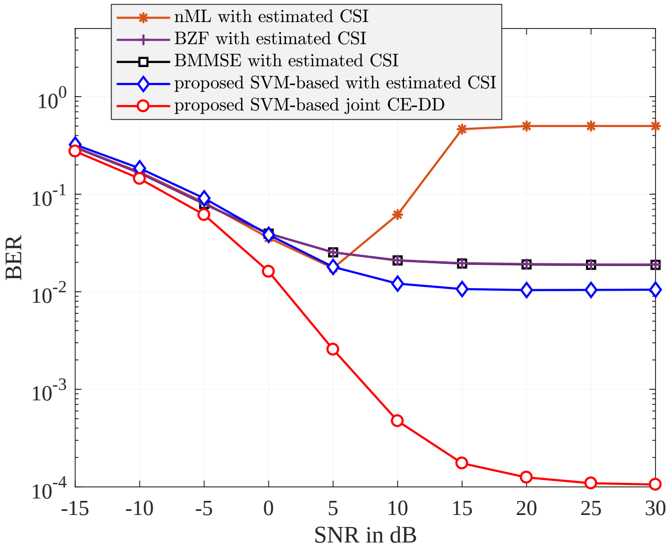


Figure 4.11: Performance comparison between two proposed data detection methods and other existing methods with estimated CSI, 16-QAM modulation, $N = 128$, $U = 8$, and $T_{\mathrm{t}} = 40$.

complexity, we are not able to provide the BER of the ML and OSD detection methods. Instead, the performance of the nML method and other linear receivers are provided as alternatives. The proposed methods not only outperform the existing methods but are also robust at high SNRs.

Finally, channel estimation and data detection results for OFDM systems with frequency-

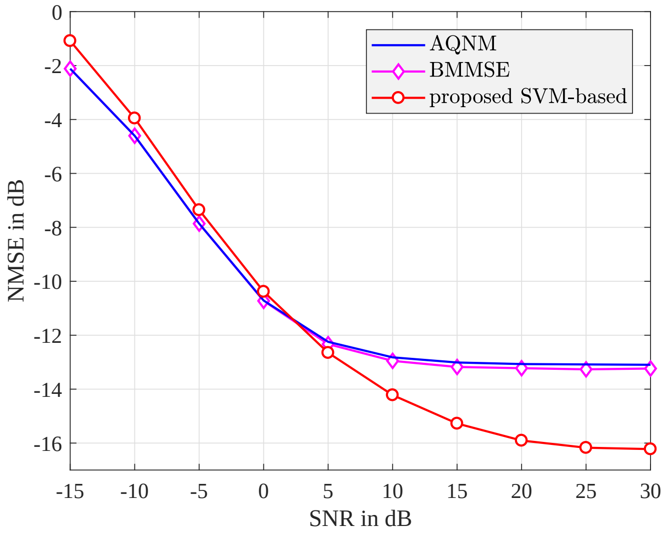


Figure 4.12: NMSE comparison between different channel estimators for an OFDM system in a frequency-selective channel with $U = 2$, $N = 16$, and $L_{\mathtt{tap}} = 8$.

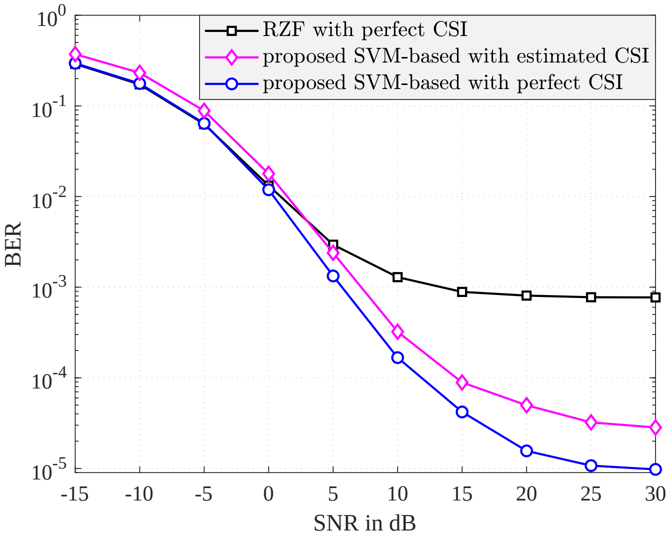


Figure 4.13: BER comparison between different data detection methods for an OFDM system in a frequency-selective channel with $N_{\mathrm{c}} = 256$, QPSK modulation, $U = 2$, $N = 16$, and $L_{\mathtt{tap}} = 8$.

selective fading channels are given in Figure 4.12 and Figure 4.13, respectively. It is observed that the BMMSE channel estimator [33] slightly outperforms the AQNM-based channel estimator [29], but both of these methods have higher NMSE than the proposed SVM-based channel estimator at high SNRs. More specifically, the high-SNR error floor of the SVM-based method is about 3-dB lower that that of the BMMSE and the AQNM-based methods.

In Figure 4.13, data detection results show that the SVM-based method considerably outperforms the Regularized Zero-Forcing (RZF) of [29]. At high SNRs, the BER of the RZF method even with perfect CSI is much higher than the BER of the SVM-based method with estimated CSI.

4.6 Conclusion

In this chapter, we have shown how linear SVM can be exploited to provide efficient and robust channel estimation and data detection. We proposed SVM-based channel estimation methods for both uncorrelated and spatially correlated channels, a two-stage SVM-based data detection method, and an SVM-based joint CE-DD method. Extension of the proposed methods to OFDM systems with frequency-selective fading channels was also derived. The key idea is to formulate the channel estimation and data detection problems as SVM problems so that they can be efficiently solved. Simulation results revealed the superiority of the proposed methods against existing ones and the gain is greatest for moderate to high SNR regimes.

Chapter 5

Deep Neural Networks for Channel Estimation and Data Detection in Low-Resolution MIMO Systems

5.1 Introduction

In the previous chapter, it has been shown that SVMs can be exploited to provide efficient and robust channel estimation and data detection in one-bit massive MIMO systems. However, the proposed SVM-based methods were specifically designed for systems with one-bit ADCs only. In this chapter, we develop a deep learning framework for channel estimation and data detection for massive MIMO systems with low-resolution ADCs. Using deep unfolding of the

The materials presented in Chapter 5 have been presented at the 2021 IEEE International Conference on Communications (ICC) in Montreal, QC, Canada [100], published in the IEEE Transactions on Wireless Communications [101], and submitted for journal publication (under 2nd round review) [102] first-order optimization iterations, we propose a channel estimator and two data detectors that are applicable for both one-bit and few-bit ADCs. The proposed channel estimation and data detection networks are model-driven and have special structures that can take advantage of domain knowledge in low-resolution MIMO systems.

We first reformulate the ML channel estimation problem by exploiting an approximation of the cdf of the normal random variable as a Sigmoid activation function. Unlike the original problem, the reformulated channel estimation approach does not lead to occasionally indeterminant gradients. Based on the reformulated problem and a deep unfolding technique, we propose a Few-Bit massive MIMO Channel Estimation Network, referred to as FBM-CENet. An interesting feature of the proposed FBM-CENet is that the pilot signal matrix is directly integrated in the weight matrices of the estimation network. When the pilot matrix is not given, it can be treated as additional trainable parameters and therefore training FBM-CENet is equivalent to *jointly optimizing both the channel estimator at the base station and the pilot signal transmitted from the users.* This is a significant advantage of the proposed FBM-CENet structure since existing channel estimators are often designed only for a known pilot matrix. The proposed DNN is based on a novel reformulation of the network layers, and is shown via several simulation results to significantly outperform the conventional DNN architecture in [53] as well as other existing channel estimation methods.

For data detection, we first propose a Bussgang decomposition-based few-bit massive MIMO Data Detection Network, referred to as B-DetNet, that is based on a linearized system model obtained through the Bussgang decomposition. Then we propose a Few-Bit massive MIMO Data Detection Network, referred to as FBM-DetNet. The special structure of FBM-DetNet is also obtained through a reformulated ML data detection problem that parallels the reformulated channel estimation problem. We stress that the proposed B-DetNet and FBM-DetNet are highly adaptive to the channel since their weight matrices and the bias vectors are defined by the channel matrix and the received signal vector, respectively. The proposed detection networks have relatively few parameters and are thus easier to train. Simulation results also show that they significantly outperform existing methods.

Next, we propose a nearest-neighbor (NN) search method to further improve the data detection performance. The idea of using two-stage detection methods has been studied previously in [31]. However, the search metric used by the second stage of [31] is susceptible to CSI errors. This issue was addressed in the previous chapter thanks to a more robust search metric. Although the second data detection stage in the previous chapter is robust, its complexity can be high like the method in [31] since the dimension of the search space over the entire candidate set can be large. The contribution of the proposed NN search method is that it generates searches over a limited number of candidates that are nearest to the solution of stage 1 and thus helps contain the search complexity. The main challenge is to obtain the set of nearest candidates efficiently and quickly. To overcome this challenge, we propose a recursive strategy that can obtain this candidate set quickly so that the proposed NN search method can be implemented in an efficient manner.

The rest of this paper is organized as follows: Section 5.2 presents the considered system model. Then, the proposed FBM-CENet is introduced in Section 5.3. The two data detection networks B-DetNet and FBM-DetNet are proposed in Section 5.4. Section 5.5 introduces the NN search method. Computational complexity analysis and numerical results are given in Section 5.6. Finally, Section 5.7 concludes the chapter.

5.2 System Model

We consider an uplink massive MIMO system with U single-antenna users and an N-antenna base station (BS), where it is assumed that $N \geq U$. Let $\mathbf{x}^{\mathbb{C}} = [x_1^{\mathbb{C}}, x_2^{\mathbb{C}}, \ldots, x_U^{\mathbb{C}}]^T \in \mathbb{C}^U$ denote the transmitted signal vector, where $x_u^{\mathbb{C}}$ is the signal transmitted from the u^{th} user. The signal $x_u^{\mathbb{C}}$ is drawn from a constellation $\mathcal{M}^{\mathbb{C}}$. Let $\mathbf{H}^{\mathbb{C}} \in \mathbb{C}^{N \times U}$ denote the channel, which is assumed to be block flat fading. Let $\mathbf{r}^{\mathbb{C}} = [r_1^{\mathbb{C}}, r_2^{\mathbb{C}}, \dots, r_N^{\mathbb{C}}]^T \in \mathbb{C}^N$ be the unquantized received signal vector at the base station, which is given as

$$\mathbf{r}^{\mathbb{C}} = \mathbf{H}^{\mathbb{C}}\mathbf{x}^{\mathbb{C}} + \mathbf{z}^{\mathbb{C}} \tag{5.1}$$

where $\mathbf{z}^{\mathbb{C}} = [z_1^{\mathbb{C}}, z_2^{\mathbb{C}}, \dots, z_N^{\mathbb{C}}]^T \in \mathbb{C}^N$ is a noise vector whose elements are assumed to be i.i.d. as $\mathcal{CN}(0, N_0)$ with noise power N_0. Each received analog signal is then quantized by a pair of b-bit ADCs to produce the quantized received signal:

$$\mathbf{y}^{\mathbb{C}} = \mathcal{Q}_b\left(\mathbf{r}^{\mathbb{C}}\right) = \mathcal{Q}_b\left(\Re\{\mathbf{r}^{\mathbb{C}}\}\right) + j\mathcal{Q}_b\left(\Im\{\mathbf{r}^{\mathbb{C}}\}\right). \tag{5.2}$$

5.3 Proposed FBM-CENet

In order to estimate the channel, a pilot sequence $\mathbf{X}_\mathrm{t}^{\mathbb{C}} \in \mathbb{C}^{U \times T_\mathrm{t}}$ of length T_t is used to generate the training data

$$\mathbf{Y}_\mathrm{t}^{\mathbb{C}} = \mathcal{Q}_b\left(\mathbf{H}^{\mathbb{C}}\mathbf{X}_\mathrm{t}^{\mathbb{C}} + \mathbf{Z}_\mathrm{t}^{\mathbb{C}}\right) \tag{5.3}$$

where $\mathbf{Y}_\mathrm{t}^{\mathbb{C}} \in \mathbb{C}^{N \times T_\mathrm{t}}$ and $\mathbf{Z}_\mathrm{t}^{\mathbb{C}} \in \mathbb{C}^{N \times T_\mathrm{t}}$. We vectorize the received signal in (5.3) to obtain

$$\mathbf{y}_\mathrm{t}^{\mathbb{C}} = \mathcal{Q}_b(\mathbf{P}^{\mathbb{C}}\mathbf{h}^{\mathbb{C}} + \mathbf{z}_\mathrm{t}^{\mathbb{C}}), \tag{5.4}$$

where $\mathbf{y}_\mathrm{t}^{\mathbb{C}} = \mathrm{vec}(\mathbf{Y}_\mathrm{t}^{\mathbb{C}})$, $\mathbf{P}^{\mathbb{C}} = (\mathbf{X}_\mathrm{t}^{\mathbb{C}})^T \otimes \mathbf{I}_N$, $\mathbf{h}^{\mathbb{C}} = \mathrm{vec}(\mathbf{H}^{\mathbb{C}})$, and $\mathbf{z}_\mathrm{t}^{\mathbb{C}} = \mathrm{vec}(\mathbf{Z}_\mathrm{t}^{\mathbb{C}})$. For convenience in later derivations, we convert the notation in (5.4) into the real domain as

$$\mathbf{y}_\mathrm{t} = \mathcal{Q}_b(\mathbf{P}\mathbf{h} + \mathbf{z}_\mathrm{t}) \tag{5.5}$$

where

$$\mathbf{y}_{\mathrm{t}} = \begin{bmatrix} \Re\{\mathbf{y}_{\mathrm{t}}^{\mathbb{C}}\} \\ \\ \Im\{\mathbf{y}_{\mathrm{t}}^{\mathbb{C}}\} \end{bmatrix}, \ \mathbf{h} = \begin{bmatrix} \Re\{\mathbf{h}^{\mathbb{C}}\} \\ \\ \Im\{\mathbf{h}^{\mathbb{C}}\} \end{bmatrix}, \text{ and } \mathbf{P} = \begin{bmatrix} \Re\{\mathbf{P}^{\mathbb{C}}\} & -\Im\{\mathbf{P}^{\mathbb{C}}\} \\ \\ \Im\{\mathbf{P}^{\mathbb{C}}\} & \Re\{\mathbf{P}^{\mathbb{C}}\} \end{bmatrix}.$$

5.3.1 Maximum-likelihood Channel Estimation Problem

Let $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{2NT_{\mathrm{t}}}]^T$, $\mathbf{y}_{\mathrm{t}} = [y_{\mathrm{t},1}, y_{\mathrm{t},2}, \ldots, y_{\mathrm{t},2NT_{\mathrm{t}}}]^T$, and $\mathbf{z}_{\mathrm{t}} = [z_{\mathrm{t},1}, \ldots, z_{\mathrm{t},2NT_{\mathrm{t}}}]^T$, then we have

$$y_{\mathrm{t},i} = \mathcal{Q}_b\left(\mathbf{p}_i^T \mathbf{h} + z_{\mathrm{t},i}\right), \quad i = 1, 2, \ldots, 2NT_{\mathrm{t}}. \tag{5.6}$$

Let $s_{\mathrm{t},i}^{\mathrm{up}} = \sqrt{2\rho}(q_{\mathrm{t},i}^{\mathrm{up}} - \mathbf{p}_i^T \mathbf{h})$ and $s_{\mathrm{t},i}^{\mathrm{low}} = \sqrt{2\rho}(q_{\mathrm{t},i}^{\mathrm{low}} - \mathbf{p}_i^T \mathbf{h})$, where $\rho = 1/N_0$ and

$$q_{\mathrm{t},i}^{\mathrm{up}} = \begin{cases} y_{\mathrm{t},i} + \frac{\Delta}{2} & \text{if } y_{\mathrm{t},i} < \tau_{2^b - 1} \\ \infty & \text{otherwise,} \end{cases} \text{ and } q_{\mathrm{t},i}^{\mathrm{low}} = \begin{cases} y_{\mathrm{t},i} - \frac{\Delta}{2} & \text{if } y_{\mathrm{t},i} > \tau_1 \\ -\infty & \text{otherwise.} \end{cases}$$

Hence, $q_{\mathrm{t},i}^{\mathrm{up}}$ and $q_{\mathrm{t},i}^{\mathrm{low}}$ are the upper and lower quantization thresholds of the bin to which $y_{\mathrm{t},i}$ belongs.

The ML channel estimator is given as follows:

$$\begin{aligned} \hat{\mathbf{h}}_{\mathrm{ML}} &= \arg\max_{\mathbf{h}} \ f(\mathbf{y}_{\mathrm{t}} \,|\, \mathbf{h}) \\ &= \arg\max_{\mathbf{h}} \sum_{i=1}^{2NT_{\mathrm{t}}} \log\left[\Phi\left(s_{\mathrm{t},i}^{\mathrm{up}}\right) - \Phi\left(s_{\mathrm{t},i}^{\mathrm{low}}\right)\right]. \end{aligned} \tag{5.7}$$

Let $\mathcal{P}_{\mathrm{t}}(\mathbf{h})$ be the objective function of (5.7). Since $\mathcal{P}_{\mathrm{t}}(\mathbf{h})$ is a concave function [103], the unconstrained optimization problem (5.7) is convex, and therefore an iterative gradient ascent

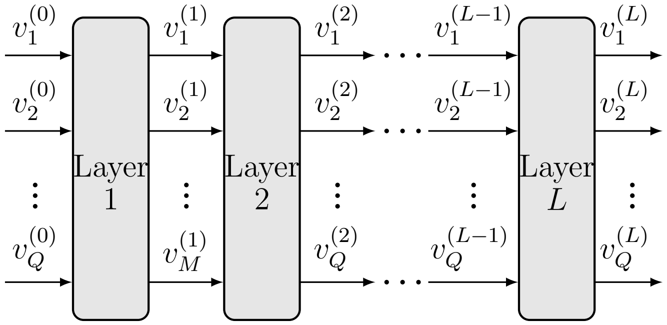


Figure 5.1: Overall structure of the proposed FBM-CENet, FBM-DetNet, and B-DetNet. For FBM-CENet, v plays the role of h and $Q = 2NU$. For FBM-DetNet and B-DetNet, v plays the role of x and $Q = 2U$.

method can be used to solve it. However, the gradient of $\mathcal{P}_{\mathrm{t}}(\mathbf{h})$, given by

$$\nabla \mathcal{P}_{\mathrm{t}}(\mathbf{h}) = \sum_{i=1}^{2NT_{\mathrm{t}}} \frac{-\sqrt{2\rho}\mathbf{p}_i\big(\phi\left(s_{\mathrm{t},i}^{\mathrm{up}}\right) - \phi\left(s_{\mathrm{t},i}^{\mathrm{low}}\right)\big)}{\Phi\left(s_{\mathrm{t},i}^{\mathrm{up}}\right) - \Phi\left(s_{\mathrm{t},i}^{\mathrm{low}}\right)}, \tag{5.8}$$

is undefined at certain points, since the function $\Phi(\cdot)$ very rapidly approaches zero or one. Specifically, as the iterative gradient descent method sequentially updates the estimated channel $\hat{\mathbf{h}}$, there exist instances of $\hat{\mathbf{h}}$ that make both $\Phi\left(s_{\mathrm{t},i}^{\mathrm{up}}\right)$ and $\Phi\left(s_{\mathrm{t},i}^{\mathrm{low}}\right)$ equal to zero or one. Thus, the denominator in (5.8) can be zero for some $\hat{\mathbf{h}}$ causing the gradient to become unbounded. In addition, a lack of a closed-form expression for $\Phi(\cdot)$ complicates the evaluation in (5.7).

These observations motivate us to reformulate the ML channel estimation problem (5.7) to address the indeterminant gradient issue as well as the complicated evaluation of the objective function in (5.7). We exploit a result in [104], which shows that the function $\Phi(t)$ can be accurately approximated by the Sigmoid function $\sigma(t) = 1/(1 + e^{-t})$ as follows:

$$\Phi(t) \approx \sigma(ct) = \frac{1}{1 + e^{-ct}} \tag{5.9}$$

where $c = 1.702$ is a constant. It was shown in [104] that $|\Phi(t) - \sigma(ct)| \leq 0.0095$, $\forall t \in \mathbb{R}$.

Using this approximation, the objective function $\mathcal{P}_{\rm t}(\mathbf{h})$ can be re-written as

$$\mathcal{P}_{\rm t}(\mathbf{h}) \approx \tilde{\mathcal{P}}_{\rm t}(\mathbf{h}) = \sum_{i=1}^{2NT_{\rm t}} \log \left[\frac{1}{1+e^{-cs_{{\rm t},i}^{\rm up}}} - \frac{1}{1+e^{-cs_{{\rm t},i}^{\rm low}}} \right] \tag{5.10}$$

and the reformulated ML channel estimation problem is

$$\hat{\mathbf{h}} = \arg\max_{\mathbf{h}} \ \tilde{\mathcal{P}}_{\rm t}(\mathbf{h}). \tag{5.11}$$

The gradient of $\tilde{\mathcal{P}}_{\rm t}(\mathbf{h})$ is

$$\begin{aligned} \nabla\tilde{\mathcal{P}}_{\rm t}(\mathbf{h}) &= \sum_{i=1}^{2NT_{\rm t}} c\sqrt{2\rho}\,\mathbf{p}_i \left(1 - \frac{1}{1+e^{cs_{{\rm t},i}^{\rm up}}} - \frac{1}{1+e^{cs_{{\rm t},i}^{\rm low}}}\right) \\ &= c\sqrt{2\rho}\,\mathbf{P}^T\Big[\mathbf{1} - \sigma\left(c\sqrt{2\rho}\,(\mathbf{Ph}-\mathbf{q}_{\rm t}^{\rm up})\right) - \sigma\left(c\sqrt{2\rho}\,(\mathbf{Ph}-\mathbf{q}_{\rm t}^{\rm low})\right)\Big] \end{aligned} \tag{5.12}$$

in which $\mathbf{q}_{\rm t}^{\rm up} = [q_{{\rm t},1}^{\rm up}, \ldots, q_{{\rm t},2NT_{\rm t}}^{\rm up}]^T$ and $\mathbf{q}_{\rm t}^{\rm low} = [q_{{\rm t},1}^{\rm low}, \ldots, q_{{\rm t},2NT_{\rm t}}^{\rm low}]^T$. Here, it should be noted that, for a matrix or vector argument, $\sigma(\cdot)$ is applied separately to every element. Unlike (5.12), it can be seen that the gradient of $\tilde{\mathcal{P}}_{\rm t}(\mathbf{h})$ does not suffer from the divide-by-zero issue. Thus, an iterative gradient descent method for solving (5.11) can be written as

$$\mathbf{h}^{(\ell)} = \mathbf{h}^{(\ell-1)} + \alpha_{\rm t}^{(\ell)}\nabla\tilde{\mathcal{P}}_{\rm t}\big(\mathbf{h}^{(\ell-1)}\big) \tag{5.13}$$

where ℓ is the iteration index and $\alpha_{\rm t}^{(\ell)}$ is the step size.

5.3.2 Structure of the proposed FBM-CENet

We employ the deep unfolding technique [105] to unfold each iteration in (5.13) as a layer of a deep neural network. The overall structure of the proposed FBM-CENet estimator is illustrated in Fig. 5.1, where each of the L layers takes a vector of $2NU$ elements as the input and generates an output vector of the same size.

The specific structure for each layer ℓ of the proposed FBM-CENet is illustrated in Fig. 5.2b. The proposed layer structure is unique due to the use of the approximation in (5.9) and the structure of the reformulated gradient in (5.12). Specifically, each layer of the proposed FBM-CENet consists of two weight matrices and two bias vectors where the pilot matrix $\mathbf{P}$ plays the role of the weight matrices and the received signals $\mathbf{q}_{\mathrm{t}}^{\mathrm{up}}$ and $\mathbf{q}_{\mathrm{t}}^{\mathrm{low}}$ play the role of the bias vectors. By contrast, each layer ℓ of a conventional DNN-based channel estimator as illustrated in Fig. 5.2a contains one weight matrix $\mathbf{W}_{\ell}$ and one bias vector $\mathbf{b}_{\ell}$. Such a conventional DNN structure has been employed in several existing works, e.g., [53–55]. An interesting feature of the proposed network is the Sigmoid activation function $\sigma(\cdot)$, which is not arbitrary but results from the use of the approximation in (5.9). This is unlike conventional DNN structures where the activation functions $\{f_{\ell}(\cdot)\}$ are often chosen heuristically by experiments.

It should be noted that the proposed FBM-CENet structure in Fig. 5.2b is free of the constant $c\sqrt{2\rho}$ since it is absorbed into the trainable parameters $\alpha_{\mathrm{t}}^{(\ell)}$ and β_{t}. If the constant is kept, each layer ℓ will contain only one trainable parameter, which is the step size $\alpha_{\mathrm{t}}^{(\ell)}$. Training $\alpha_{\mathrm{t}}^{(\ell)}$ can be interpreted as moving along the gradient directions and optimizing the step size at each layer. We refer this network structure to as purely gradient-based FBM-CENet (PG-FBM-CENet). Since different values of β_{t} result in different directions in the vicinity of the gradient, training FBM-CENet can be interpreted as jointly learning the optimal directions and the associated optimal step sizes. This helps FBM-CENet improve the performance compared to PG-FBM-CENet. The reason is that always moving along the gradient direction may not be optimal. FBM-CENet learns an optimal path that makes the network output (the channel estimate) closer to the true channel vector. We will numerically show that FBM-CENet outperforms PG-FBM-CENet later.

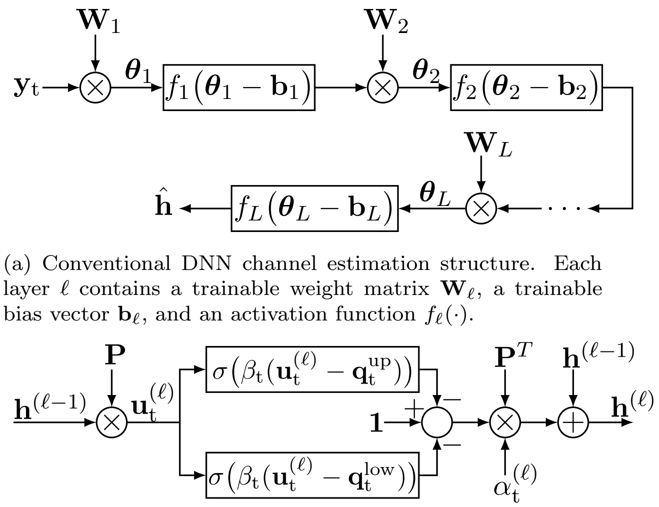
5.3.3 Trainable parameters

For a given pilot matrix $\mathbf{P}$, the trainable parameters in the proposed FBM-CENet are the step sizes $\{\alpha_{\mathrm{t}}^{(\ell)}\}$ and the scaling parameter β_{t} inside the Sigmoid function. Note that as mentioned above, the coefficient $c\sqrt{2\rho}$ is omitted in the proposed network since it is absorbed in the trainable parameters $\{\alpha_{\mathrm{t}}^{(\ell)}\}$ and β_{t}.

It is important to note that the pilot matrix $\mathbf{P}$ directly plays the role of the weight matrices. Therefore, when the pilot matrix $\mathbf{P}$ is not given, it too can be treated as a trainable parameter. In this case, training the proposed FBM-CENet is equivalent to *jointly optimizing both the channel estimator at the base station and the pilot signal transmitted from the users.* This is a significant advantage of the proposed network structure since the conventional DNN-based channel estimator is often trained or optimized for a given pilot matrix, and thus is unable to convey information about the optimal pilot signal. The approach proposed in [53] also jointly optimized the pilot signal and the channel estimator for massive MIMO systems with low-resolution ADCs, but it employs the conventional DNN structure illustrated in Fig. 5.2a. We will later show that the proposed FBM-CENet estimator significantly outperforms the method in [53].

5.3.4 Training strategy

Here we present the strategy for training the proposed FBM-CENet estimator. Let $\hat{\mathbf{h}}$ denote the channel estimate, which is set to be the output of the last layer of FBM-CENet, i.e., $\hat{\mathbf{h}} = \mathbf{h}^{(L)}$. The cost function to be minimized is $\|\hat{\mathbf{h}} - \mathbf{h}\|^2$. We choose this cost function instead of the objective function in (5.10) because the value of (5.10) is undefined when the argument of the logarithm approaches zero. In our investigation, training using the cost function (5.10) was not successful due to this issue. When the pilot matrix $\mathbf{P}$ is given, a training sample for FBM-CENet consists of the given matrix $\mathbf{P}$, a channel vector realization



$\mathbf{W}_1$
$\mathbf{W}_2$
$\mathbf{y}_\mathrm{t}$
$\boldsymbol{\theta}_1$
$f_1(\boldsymbol{\theta}_1 - \mathbf{b}_1)$
$\boldsymbol{\theta}_2$
$f_2(\boldsymbol{\theta}_2 - \mathbf{b}_2)$
$\mathbf{W}_L$
$\hat{\mathbf{h}}$
$f_L(\boldsymbol{\theta}_L - \mathbf{b}_L)$
$\boldsymbol{\theta}_L$
$\cdots$
$\mathbf{P}$
$\mathbf{h}^{(\ell-1)}$
$\mathbf{u}_\mathrm{t}^{(\ell)}$
$\sigma\big(\beta_\mathrm{t}(\mathbf{u}_\mathrm{t}^{(\ell)} - \mathbf{q}_\mathrm{t}^\mathrm{up})\big)$
$\sigma\big(\beta_\mathrm{t}(\mathbf{u}_\mathrm{t}^{(\ell)} - \mathbf{q}_\mathrm{t}^\mathrm{low})\big)$
$\mathbf{1}$
$+$
$-$
$-$
$\mathbf{P}^T$
$\mathbf{h}^{(\ell-1)}$
$\alpha_\mathrm{t}^{(\ell)}$
$\mathbf{h}^{(\ell)}$

(a) Conventional DNN channel estimation structure. Each layer ℓ contains a trainable weight matrix $\mathbf{W}_\ell$, a trainable bias vector $\mathbf{b}_\ell$, and an activation function $f_\ell(\cdot)$.

(b) Specific structure of layer ℓ of the proposed FBM-CENet.

Figure 5.2: Conventional versus proposed DNN structure for channel estimation.

$\mathbf{h}$ and a Gaussian noise vector $\mathbf{z}$, which can be randomly generated. When the pilot matrix $\mathbf{P}$ is not given and is trainable, a training sample only consists of $\mathbf{h}$ and a Gaussian noise vector $\mathbf{z}$. Note that $\mathbf{h}$ is randomly generated according to a particular channel model.

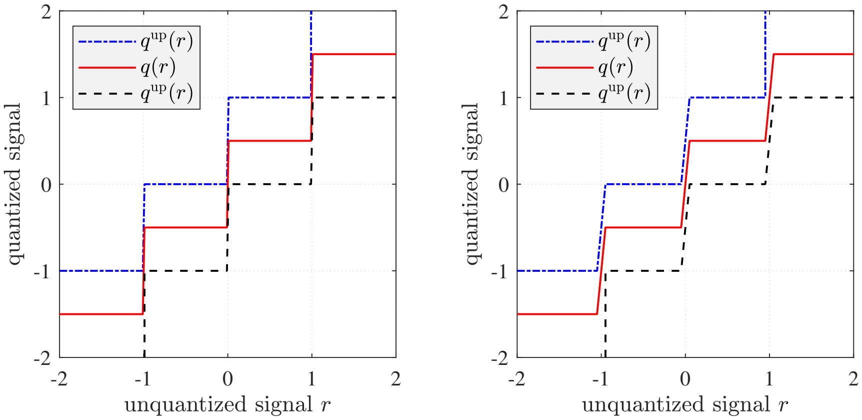
It is important to note that the received signals $\mathbf{q}_\mathrm{t}^\mathrm{up}$ and $\mathbf{q}_\mathrm{t}^\mathrm{low}$ depend on the pilot matrix $\mathbf{P}$. Therefore, when the pilot matrix $\mathbf{P}$ is trainable, gradient back-propagation during the training process should also go through $\mathbf{q}_\mathrm{t}^\mathrm{up}$ and $\mathbf{q}_\mathrm{t}^\mathrm{low}$. However, the low-resolution ADCs are discontinuous functions, which make gradient back-propagation through $\mathbf{q}_\mathrm{t}^\mathrm{up}$ and $\mathbf{q}_\mathrm{t}^\mathrm{low}$ infeasible. To overcome this issue, we employ a soft quantizer model based on the Rectified Linear Unit (ReLU) function $f_\mathrm{relu}(r) = \max(0, r)$ for the training process as follows:

$$q^\mathrm{up}(r) = q(r) + \frac{\Delta}{2} + c_2\big[f_\mathrm{relu}(r - B\Delta + c_1) - f_\mathrm{relu}(r - B\Delta - c_1)\big] \tag{5.14}$$

$$q^\mathrm{low}(r) = q(r) - \frac{\Delta}{2} - c_2\big[f_\mathrm{relu}(-r - B\Delta + c_1) - f_\mathrm{relu}(-r - B\Delta - c_1)\big] \tag{5.15}$$

where $B = 2^{b-1} - 1$, c_1 and c_2 are positive constants, and

$$q(r) = -(2^b - 1)\frac{\Delta}{2} + \frac{\Delta}{2c_1}\sum_{i=-B}^{B}\big[f_\mathrm{relu}(r + i\Delta + c_1) - f_\mathrm{relu}(r + i\Delta - c_1)\big]. \tag{5.16}$$



(a) $c_1 = 0.01$.

(b) $c_1 = 0.05$.

Figure 5.3: Two-bit relu-based soft quantizer with $\Delta = 1$.

The resulting ReLU-based function is continuous and therefore back-propagation is feasible. The effect of c_1 is illustrated in Fig. 5.3. It can be seen that smaller values of c_1 make the soft quantizer sharper, or in other words closer to the actual hard quantizer. The constant c_2 accounts for the two thresholds $\tau_0 = -\infty$ and $\tau_{2^b} = \infty$, and hence should be large. The constants $\{c_1, c_2\}$ should not be treated as trainable parameters because it is necessary for the soft quantizer to be a reasonable approximation of the hard quantizer. Allowing these constants to be trained may produce a large deviation between the soft and hard quantizers.

Note that the soft quantizer could also be modeled using the tanh function as follows:

$$q^{\mathrm{up}}(r) = q(r) + \frac{\Delta}{2} + c_4 f_{\tanh}(c_3(r - B\Delta)) \tag{5.17}$$

$$q^{\mathrm{low}}(r) = q(r) - \frac{\Delta}{2} - c_4 f_{\tanh}(c_3(-r - B\Delta)) \tag{5.18}$$

where $f_{\tanh}(r) = (\tanh(r) + 1)/2$ and

$$q(r) = \frac{\Delta}{2}\left[f_{\tanh}(c_3 r) - f_{\tanh}(-c_3 r)\right] + \Delta \sum_{i=1}^{B} f_{\tanh}(c_3(r - i\Delta)) - f_{\tanh}(c_3(-r - i\Delta)). \tag{5.19}$$

Larger values of c_3 make the soft quantizer sharper. The constant c_4 accounts for the two thresholds τ_0 and τ_{2^b}, and hence should also be large. Although we implement our networks with the ReLU-based model, in the simulations we will show that both the tanh- and ReLU-based soft quantizers yield essentially the same performance.

5.4 Data Detection in Few-Bit MIMO Systems

In this section, we propose two DNN-based detectors, referred to as B-DetNet and FBM-DetNet, for low-resolution massive MIMO systems. For convenience in later derivations, we convert (5.1) and (5.2) into the real domain as follows:

$$\mathbf{y} = \mathcal{Q}_b\left(\mathbf{H}\mathbf{x} + \mathbf{z}\right), \tag{5.20}$$

where

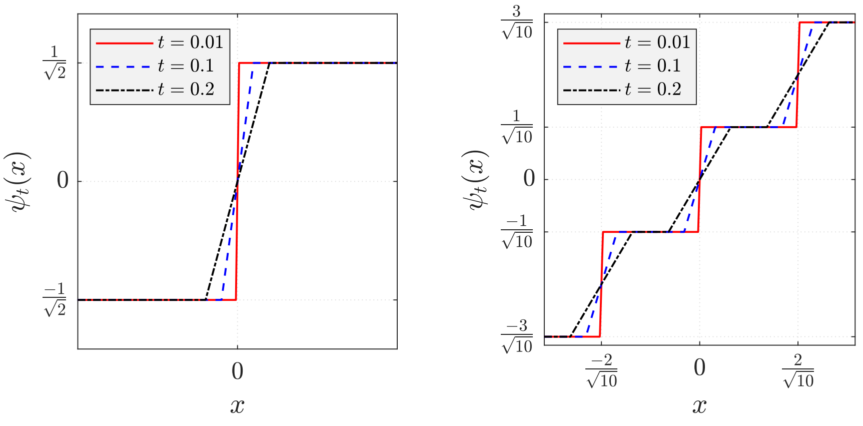
$$\mathbf{y} = \begin{bmatrix} \Re\{\mathbf{y}^{\mathbb{C}}\} \\ \Im\{\mathbf{y}^{\mathbb{C}}\} \end{bmatrix}, \ \mathbf{x} = \begin{bmatrix} \Re\{\mathbf{x}^{\mathbb{C}}\} \\ \Im\{\mathbf{x}^{\mathbb{C}}\} \end{bmatrix}, \ \mathbf{z} = \begin{bmatrix} \Re\{\mathbf{z}^{\mathbb{C}}\} \\ \Im\{\mathbf{z}^{\mathbb{C}}\} \end{bmatrix}, \text{ and } \mathbf{H} = \begin{bmatrix} \Re\{\mathbf{H}^{\mathbb{C}}\} & -\Im\{\mathbf{H}^{\mathbb{C}}\} \\ \Im\{\mathbf{H}^{\mathbb{C}}\} & \Re\{\mathbf{H}^{\mathbb{C}}\} \end{bmatrix}.$$

Note that $\mathbf{y} \in \mathbb{R}^{2N}$, $\mathbf{x} \in \mathbb{R}^{2U}$, $\mathbf{z} \in \mathbb{R}^{2N}$, and $\mathbf{H} \in \mathbb{R}^{2N \times 2U}$. We also denote $\mathbf{y} = [y_1, \ldots, y_{2N}]^T$ and $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_{2N}]^T$.

5.4.1 Proposed B-DetNet

Applying the Bussgang decomposition to (5.20), we obtain

$$\begin{aligned} \mathbf{y} &= \mathbf{V}\mathbf{H}\mathbf{x} + \mathbf{V}\mathbf{z} + \mathbf{d} \\ &= \mathbf{A}\mathbf{x} + \mathbf{n} \end{aligned} \tag{5.21}$$



(a) QPSK signaling.

(b) 16QAM signaling.

Figure 5.4: Projector function $\psi_t(\cdot)$ with different values of t.

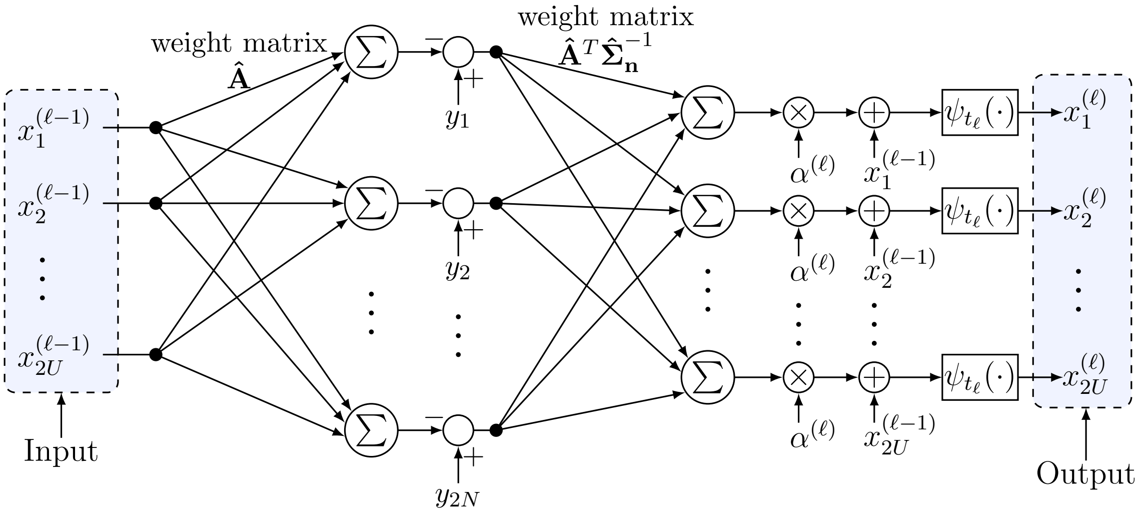


Figure 5.5: Specific structure of layer ℓ of the proposed B-DetNet.

where $\mathbf{V} \in \mathbb{R}^{2N \times 2N}$ is a diagonal matrix and given as

$$\mathbf{V} = \frac{\Delta}{\sqrt{2\pi}} \operatorname{diag}(\mathbf{\Sigma}_{\mathbf{r}})^{-\frac{1}{2}} \times \sum_{i=1}^{2^b-1} \exp\left\{ -\frac{1}{2}\Delta^2 (i - 2^{b-1})^2 \operatorname{diag}(\mathbf{\Sigma}_{\mathbf{r}})^{-1} \right\}$$

and $\mathbf{\Sigma_r} = \mathbf{H}\mathbf{\Sigma_x}\mathbf{H}^T + \frac{N_0}{2}\mathbf{I} \in \mathbb{R}^{2N\times 2N}$. For the case of 1-bit ADCs, the covariance of $\mathbf{n}$ is given in closed form as [20]

$$\begin{aligned}\mathbf{\Sigma_n} =& \frac{\Delta^2}{2\pi}\Big[\arcsin\Big(\operatorname{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}}\mathbf{\Sigma_r}\operatorname{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}}\Big) - \\ &\operatorname{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}}\mathbf{\Sigma_r}\operatorname{diag}(\mathbf{\Sigma_r})^{-\frac{1}{2}} + \frac{N_0}{2}\operatorname{diag}(\mathbf{\Sigma_r})^{-1}\Big].\end{aligned} \tag{5.22}$$

For few-bit ADCs, the covariance of $\mathbf{n}$ can be approximated as $\mathbf{\Sigma_n} \approx \frac{N_0}{2}\mathbf{V}\mathbf{V}^T + \eta_b \operatorname{diag}(\mathbf{\Sigma_r})$. The effective noise $\mathbf{n}$ is often modeled as $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma_n})$. Based on this linearized model, different linear detectors such as BZF, BMMSE, and BWZF were introduced in [52, 64].

Here, we propose the data detection network B-DetNet based on the linearized system model in (5.21). Since the effective noise $\mathbf{n}$ is assumed to be Gaussian, the Bussgang-based maximum likelihood detection problem is given as

$$\hat{\mathbf{x}}_{\texttt{BML}} = \underset{\mathbf{x}^{\mathbb{C}} \in (\mathcal{M}^{\mathbb{C}})^U}{\arg\min}\ (\mathbf{y} - \mathbf{A}\mathbf{x})^T \mathbf{\Sigma_n}^{-1} (\mathbf{y} - \mathbf{A}\mathbf{x}). \tag{5.23}$$

Let $P_{\text{B}}(\mathbf{x})$ be the objective function of (5.23). Note that $P_{\text{B}}(\mathbf{x})$ is a quadratic function of $\mathbf{x}$ and thus convex, but the optimization problem is not convex due to the discrete feasibility constraint $\mathbf{x}^{\mathbb{C}} \in (\mathcal{M}^{\mathbb{C}})^U$. An optimal solution to (5.23) therefore requires an exhaustive search, which is very expensive for large scale systems. Instead, an iterative projected gradient descent method

$$\mathbf{x}^{(\ell)} = \psi_{t_\ell}\left(\mathbf{x}^{(\ell-1)} - \alpha^{(\ell)}\nabla P_{\text{B}}(\mathbf{x}^{(\ell-1)})\right) \tag{5.24}$$

can be applied to search for the optimal solution. Herein, the gradient of $P_{\text{B}}(\mathbf{x})$ evaluated at $\mathbf{x}^{(\ell-1)}$ is given by

$$\nabla P_{\text{B}}(\mathbf{x}^{(\ell-1)}) = -2\mathbf{A}^T\mathbf{\Sigma_n}^{-1}\left(\mathbf{y} - \mathbf{A}\mathbf{x}^{(\ell-1)}\right) \tag{5.25}$$

and $\psi_{t_\ell}(\cdot)$, characterized by the positive parameter t_ℓ, is a non-linear projector that forces the signal to the nearest constellation point. Based on the ReLU activation function $q(r)$ in (5.16), $\psi_{t_\ell}(\cdot)$ can be written as

$$\psi_{t_\ell}(x) = -(2^{b'} - 1)\frac{\Delta'}{2} + \frac{\Delta'}{2t_\ell} \sum_{i=-B'}^{B'} \left[f_{\mathrm{relu}}(r + i\Delta + t_\ell) - f_{\mathrm{relu}}(r + i\Delta - t_\ell) \right] \tag{5.26}$$

where $B' = 2^{b'-1} - 1$. For QPSK signalling, $\{b', \Delta'\} = \{1, \frac{2}{\sqrt{2}}\}$ and for 16-QAM signalling, $\{b', \Delta'\} = \{2, \frac{2}{\sqrt{10}}\}$. The effect of t_ℓ on $\psi_t(\cdot)$ is illustrated in Fig. 5.4. It can be seen that a smaller t_ℓ also makes the projector sharper. Such a projection function was used in [106], which studied deep learning-based detection for unquantized MIMO systems.

Our proposed B-DetNet approach is realized by unfolding the projected gradient descent in (5.24). The overall structure of B-DetNet is illustrated in Fig. 5.1. Each layer takes an input vector of size $2U$ and generates an output vector of the same size. The specific structure of each B-DetNet layer is given in Fig. 5.5 where $\hat{\mathbf{A}}$ and $\hat{\mathbf{A}}^T \hat{\mathbf{\Sigma}}_{\mathbf{n}}^{-1}$ play the role of weight matrices. Note that $\hat{\mathbf{A}}$ and $\hat{\mathbf{\Sigma}}_{\mathbf{n}}^{-1}$ are obtained using a channel estimate $\hat{\mathbf{H}}$ from, e.g., FBM-CENet. The received signal vector $\mathbf{y}$ is seen to be the bias vector. Hence, B-DetNet is highly adaptive to the channel. The only trainable parameters in layer ℓ of B-DetNet are the step size $\alpha^{(\ell)}$ and the scaling parameter t_ℓ in the projector function $\psi_{t_\ell}(\cdot)$.

We note that similar structures for data detection in full-resolution systems have been developed in [106, 107]. However, the received signal in full-resolution systems is given as $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{z}$, and therefore the gradient of interest becomes $-2\mathbf{H}^T(\mathbf{y} - \mathbf{H}\mathbf{x})$. For low-resolution systems, we have a new effective channel $\mathbf{A}$ and a new noise covariance matrix $\mathbf{\Sigma}_{\mathbf{n}}$, resulting in a new form of the gradient as in (5.25).

5.4.2 Proposed FBM-DetNet

Maximum-likelihood data detection problem:

Let $s_i^{\text{up}} = \sqrt{2\rho}(q_i^{\text{up}} - \mathbf{h}_i^T\mathbf{x})$ and $s_i^{\text{low}} = \sqrt{2\rho}(q_i^{\text{low}} - \mathbf{h}_i^T\mathbf{x})$, where

$$q_i^{\text{up}} = \begin{cases} y_i + \frac{\Delta}{2} & \text{if } y_i < \tau_{2^b-1} \\ \infty & \text{otherwise,} \end{cases} \text{ and } q_i^{\text{low}} = \begin{cases} y_i - \frac{\Delta}{2} & \text{if } y_i > \tau_1 \\ -\infty & \text{otherwise.} \end{cases}$$

Hence, q_i^{up} and q_i^{low} are the upper and lower quantization thresholds of the bin to which y_i belongs. The ML detection problem based on the log-likelihood function for the model in (5.20) is defined as follows [60]:

$$\hat{\mathbf{x}}_{\text{ML}} = \underset{\mathbf{x}^{\mathbb{C}}\in(\mathcal{M}^{\mathbb{C}})^U}{\arg\max} \sum_{i=1}^{2N} \log\left[\Phi\left(s_i^{\text{up}}\right) - \Phi\left(s_i^{\text{low}}\right)\right]. \tag{5.27}$$

Let $\mathcal{P}(\mathbf{x})$ denote the objective function of (5.27), which is a concave function of $\mathbf{x}$. However, the optimization problem (5.27) is not convex since the feasible set is discrete. Therefore, an optimal solution for ML detection in (5.27) also requires an exhaustive search over $\mathbf{x}^{\mathbb{C}} \in (\mathcal{M}^{\mathbb{C}})^U$, which is prohibitively complex for large-scale systems. One can relax the constraint on the feasible set from $\mathbf{x}^{\mathbb{C}} \in (\mathcal{M}^{\mathbb{C}})^U$ to $\mathbf{x}^{\mathbb{C}} \in \mathbb{C}^U$ in order to obtain a convex optimization problem and allow an iterative gradient descent method to be used. Unfortunately, such an approach also suffers from the indeterminant gradient issue discussed earlier for the channel estimation problem. In addition, there is no closed-form expression for $\Phi(\cdot)$, which complicates the evaluation in (5.27). As before, we exploit the approximation in (5.9) to obtain an approximate version of the function $\mathcal{P}(\mathbf{x})$ as follows:

$$\mathcal{P}(\mathbf{x}) \approx \tilde{\mathcal{P}}(\mathbf{x}) = \sum_{i=1}^{2N} \log\left[\frac{1}{1+e^{-cs_i^{\text{up}}}} - \frac{1}{1+e^{-cs_i^{\text{low}}}}\right] \tag{5.28}$$

The reformulated ML detection problem is thus

$$\hat{\mathbf{x}}_{\mathtt{ML}} = \underset{\mathbf{x}^{\mathbb{C}} \in (\mathcal{M}^{\mathbb{C}})^{U}}{\arg\max}\ \tilde{\mathcal{P}}(\mathbf{x}), \tag{5.29}$$

and the gradient of $\tilde{\mathcal{P}}(\mathbf{x})$ is

$$\begin{aligned} \nabla\tilde{\mathcal{P}}(\mathbf{x}) &= \sum_{i=1}^{2N} c\sqrt{2\rho}\,\mathbf{h}_i \left(1 - \frac{1}{1+e^{cs_i^{\text{up}}}} - \frac{1}{1+e^{cs_i^{\text{low}}}}\right) && (5.30) \\ &= c\sqrt{2\rho}\,\mathbf{H}^T \Big[\mathbf{1} - \sigma\left(c\sqrt{2\rho}\,(\mathbf{H}\mathbf{x} - \mathbf{q}^{\text{up}})\right) - \sigma\left(c\sqrt{2\rho}\,(\mathbf{H}\mathbf{x} - \mathbf{q}^{\text{low}})\right)\Big] && (5.31) \end{aligned}$$

where $\mathbf{q}^{\text{up}} = [q_1^{\text{up}}, \ldots, q_{2N}^{\text{up}}]^T$ and $\mathbf{q}^{\text{low}} = [q_1^{\text{low}}, \ldots, q_{2N}^{\text{low}}]^T$. Thus, an iterative projected gradient decent method for solving (5.29) can be written as

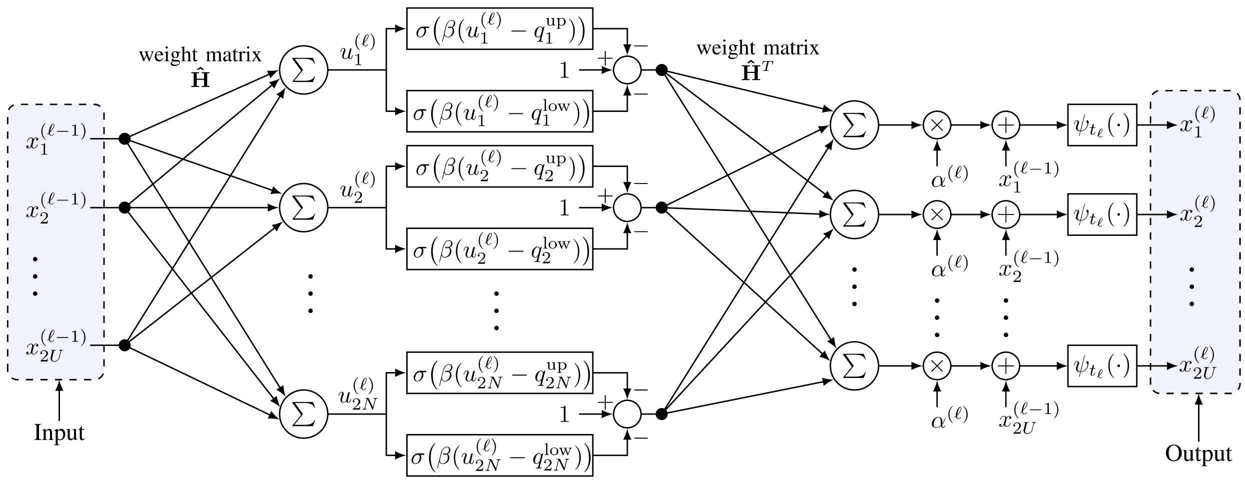
$$\mathbf{x}^{(\ell)} = \psi_{t_\ell}\left(\mathbf{x}^{(\ell-1)} + \alpha^{(\ell)}\nabla\tilde{\mathcal{P}}(\mathbf{x}^{(\ell-1)})\right) \tag{5.32}$$

where ℓ is the iteration index, $\alpha^{(\ell)}$ is a step size, and $\psi_{t_\ell}(\cdot)$ is also a projector as defined in (5.26).

Structure of the proposed FBM-DetNet

In order to optimize the step sizes $\{\alpha^{(\ell)}\}$ and scaling parameters $\{t_\ell\}$ of the projection function, we also unfold each iteration in (5.32) as a separate DNN layer. The overall structure of the proposed detector FBM-DetNet is also illustrated in Fig. 5.1, and is similar to that of B-DetNet as each layer of both networks takes a vector of $2U$ elements as the input and generates an output vector of the same size.

The specific structure for each layer ℓ of FBM-DetNet is illustrated in Fig. 5.6. Each layer of FBM-DetNet has two weight matrices $\mathbf{H}$ and $\mathbf{H}^T$ and two bias vectors $\mathbf{q}^{\text{up}}$ and $\mathbf{q}^{\text{low}}$ defined by the channel and the received signal, respectively. The activation function is the Sigmoid



$x_1^{(\ell-1)}$
$x_2^{(\ell-1)}$
$x_{2U}^{(\ell-1)}$
Input
weight matrix
$\hat{\mathbf{H}}$
Σ
$u_1^{(\ell)}$
$u_2^{(\ell)}$
$u_{2N}^{(\ell)}$
$\sigma\big(\beta(u_1^{(\ell)} - q_1^{\mathrm{up}})\big)$
$\sigma\big(\beta(u_1^{(\ell)} - q_1^{\mathrm{low}})\big)$
$\sigma\big(\beta(u_2^{(\ell)} - q_2^{\mathrm{up}})\big)$
$\sigma\big(\beta(u_2^{(\ell)} - q_2^{\mathrm{low}})\big)$
$\sigma\big(\beta(u_{2N}^{(\ell)} - q_{2N}^{\mathrm{up}})\big)$
$\sigma\big(\beta(u_{2N}^{(\ell)} - q_{2N}^{\mathrm{low}})\big)$
1
+
−
weight matrix
$\hat{\mathbf{H}}^T$
$\alpha^{(\ell)}$
$x_1^{(\ell-1)}$
$x_2^{(\ell-1)}$
$x_{2U}^{(\ell-1)}$
$\psi_{t_\ell}(\cdot)$
$x_1^{(\ell)}$
$x_2^{(\ell)}$
$x_{2U}^{(\ell)}$
Output

Figure 5.6: Specific structure of layer ℓ of FBM-DetNet. The weight matrices and the bias vectors are defined by the channel and the received signal, respectively.

function $\sigma(\cdot)$ due to the use of the approximation in (5.9). Since $\mathbf{H} \in \mathbb{R}^{2N\times 2U}$, the learning process for each layer of FBM-DetNet can be interpreted as first up-converting the signal $\mathbf{x}^{(\ell-1)}$ from dimension $2U$ to dimension $2N$ using the weight matrix $\mathbf{H}$, then applying the nonlinear activation function $\sigma(\cdot)$ before down-converting the signal back to dimension $2U$ using the weight matrix $\mathbf{H}^T$. Finally, the function $\psi_{t_\ell}(\cdot)$ is implemented to project $\mathbf{x}^{(\ell-1)}$ onto the discrete set $(\mathcal{M}^{\mathbb{C}})^U$.

The layers of FBM-DetNet are similar to those of FBM-CENet in Fig. 5.2b. However, while the weight matrices of FBM-CENet are defined by the pilot matrix $\mathbf{P}$ which is trainable, the weight matrices of FBM-DetNet are defined by the channel matrix $\mathbf{H}$ which is not. Thus, FBM-DetNet is highly adaptive to the channel. The trainable parameters of FBM-DetNet are the step sizes $\{\alpha^{(\ell)}\}$, scaling parameters $\{t_\ell\}$ for the projector, and a scaling parameter β for the Sigmoid function. Note that the coefficient $c\sqrt{2\rho}$ is also omitted in FBM-DetNet for the same reason as in FBM-CENet.

5.4.3 Training strategy

A training sample for the two proposed data detection networks, B-DetNet and FBM-DetNet, can be obtained by first randomly generating a channel matrix $\mathbf{H}$ according to a particular channel model, then obtaining an estimate $\hat{\mathbf{H}}$ of $\mathbf{H}$ by using, e.g., FBM-CENet. Next, the transmit signal $\mathbf{x}$ can be randomly drawn from the signal constellation and a Gaussian noise vector $\mathbf{z}$ can be chosen to obtain a representative received signal vector $\mathbf{y} = \mathcal{Q}_b(\mathbf{Hx} + \mathbf{z})$. Similar to the channel estimation network, the cost function to be minimized is $\|\mathbf{x}^{(L)} - \mathbf{x}\|^2$, where $\mathbf{x}$ is the target (transmitted) signal. For training the proposed data detection networks, we do not need to use the soft quantization model because the trainable parameters do not appear in the received signals $\mathbf{y}$ or $\mathbf{q}^{\text{up}}$ and $\mathbf{q}^{\text{low}}$, and therefore the exact hard quantizer can be used.

5.5 Nearest-Neighbor Search for Second-Stage Detection

In this section, an NN search method, operating as a second data detection stage, is proposed to further improve the data detection performance. The idea of using two-stage data detection has already been used in [31]. However, the search space can be very large when the number of users is large, and so not efficient in terms of computational complexity. Let $\tilde{\mathbf{x}}$ denote an estimate of the transmitted signal, e.g., given in stage 1 by B-DetNet, FBM-DetNet, or other detectors, the proposed NN search method first finds a limited set of symbol vectors that are nearest to $\tilde{\mathbf{x}}$ and then searches over that set for the most likely symbol vector as the final detection solution. The contribution of the proposed NN search method is that it generates searches over a limited number of symbol vectors that are nearest to the estimate $\tilde{\mathbf{x}}$, and thus significantly reduces the computational load.

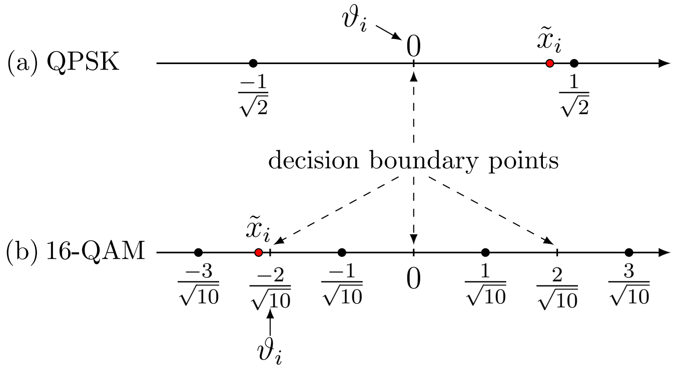


Figure 5.7: An example for the relative difference between $\tilde{x}_i$ and the constellation points: (a) the estimate $\tilde{x}_i$ is far from $\vartheta_i = 0$ and close to the constellation point $1/\sqrt{2}$, which means there is a high probability that the transmitted signal x_i is $1/\sqrt{2}$; (b) the estimate $\tilde{x}_i$ is close to the boundary point $\vartheta_i = -2/\sqrt{10}$, thus it is difficult to say if $-3/\sqrt{10}$ or $-1/\sqrt{10}$ was transmitted.

We denote $\mathcal{M}$ as the constellation in the real domain; for example, $\mathcal{M} = \left\{\pm\frac{1}{\sqrt{2}}\right\}$ for QPSK and $\mathcal{M} = \left\{\pm\frac{1}{\sqrt{10}}, \pm\frac{3}{\sqrt{10}}\right\}$ for 16-QAM. Let $\mathcal{B}$ be the set of decision boundary points; *i.e.,* $\mathcal{B} = \{0\}$ for QPSK and $\mathcal{B} = \left\{0, \pm\frac{2}{\sqrt{10}}\right\}$ for 16-QAM. Denote $\tilde{\mathbf{x}} = [\tilde{x}_1, \ldots, \tilde{x}_{2U}]^T$ and $\boldsymbol{\vartheta} = [\vartheta_1, \ldots, \vartheta_{2U}]^T$, where ϑ_i is the decision boundary point that is nearest to $\tilde{x}_i$, as follows:

$$\vartheta_i = \arg\min_{\vartheta \in \mathcal{B}} |\vartheta - \tilde{x}_i|, \quad i \in \{1, 2, \ldots, 2U\}. \tag{5.33}$$

An illustrative example for the relative difference between $\tilde{x}_i$ and the constellation points is given in Fig. 5.7. This example illustrates the problem that occurs when $\tilde{x}_i$ is close to a decision boundary point, where symbol-by-symbol detection may not be reliable. Here, we use a threshold $\gamma > 0$ to classify whether symbol-by-symbol detection is used or not. More specifically, if the distance from $\tilde{x}_i$ to its nearest decision boundary point ϑ_i is greater than γ, i.e., $|\tilde{x}_i - \vartheta_i| > \gamma$, then we can use symbol-by-symbol detection for $\tilde{x}_i$. When $|\tilde{x}_i - \vartheta_i| \leq \gamma$, symbol-by-symbol detection is not reliable, and so we list the two nearest constellation points to $\tilde{x}_i$ as the candidates for the transmitted signal x_i.

Let $\mathcal{A}_i$ denote the set of candidates for the transmitted signal x_i. When $|\tilde{x}_i - \vartheta_i| > \gamma$, we apply symbol-by-symbol detection and so

$$\mathcal{A}_i = \left\{ \underset{x \in \mathcal{M}}{\arg\min}\ |x - \tilde{x}_i| \right\}.$$

When $|\tilde{x}_i - \vartheta_i| \le \gamma$, we have $\mathcal{A}_i = \left\{\vartheta_i \pm \frac{1}{\sqrt{2}}\right\} = \left\{\pm\frac{1}{\sqrt{2}}\right\}$ for QPSK and $\mathcal{A}_i = \left\{\vartheta_i \pm \frac{1}{\sqrt{10}}\right\}$ for 16-QAM. Hence, $\mathcal{A}_i$ contains only one or two elements. The following example illustrates the formation of $\mathcal{A}_i$.

Example 5.1. *Suppose that* $\tilde{\mathbf{x}} = [0.1, -0.5, -0.3, 0.8]^T$ *and QPSK modulation is used with* $\gamma = \frac{1}{2\sqrt{2}} \approx 0.35$. *Note here that* $\vartheta_1 = \vartheta_2 = \vartheta_3 = \vartheta_4 = 0$. *We have*

- $\mathcal{A}_1 = \mathcal{A}_3 = \{\pm\frac{1}{\sqrt{2}}\}$ *because* $|\tilde{x}_1 - \vartheta_1| = 0.1 < \gamma$ *and* $|\tilde{x}_3 - \vartheta_3| = 0.3 < \gamma$,
- $\mathcal{A}_2 = \{\frac{-1}{\sqrt{2}}\}$ *because* $|\tilde{x}_2 - \vartheta_2| = 0.5 > \gamma$ *and* $\tilde{x}_2$ *is closer to* $\frac{-1}{\sqrt{2}}$ *than* $\frac{1}{\sqrt{2}}$, *i.e.,* $\left|\tilde{x}_2 - \frac{-1}{\sqrt{2}}\right| < \left|\tilde{x}_2 - \frac{1}{\sqrt{2}}\right|$,
- $\mathcal{A}_4 = \{\frac{1}{\sqrt{2}}\}$ *because* $|\tilde{x}_4 - \vartheta_4| = 0.8 > \gamma$ *and* $\tilde{x}_4$ *is closer to* $\frac{1}{\sqrt{2}}$ *than* $\frac{-1}{\sqrt{2}}$, *i.e.,* $\left|\tilde{x}_4 - \frac{1}{\sqrt{2}}\right| < \left|\tilde{x}_4 - \frac{-1}{\sqrt{2}}\right|$.

Hence, in this example, $\mathcal{A}_1$ *and* $\mathcal{A}_3$ *have two elements while* $\mathcal{A}_2$ *and* $\mathcal{A}_4$ *have only one element.*

The complete set of candidates for the transmitted signal vector is given by the Cartesian product

$$\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \ldots \times \mathcal{A}_{2U},$$

and so the size of $\mathcal{A}$ is $|\mathcal{A}| = \prod_{i=1}^{2U} |\mathcal{A}_i| = 2^A$, where A is the number of sets $\mathcal{A}_i$ having two elements. The search methods in [31] and in Chapter 4 always search over the entire set $\mathcal{A}$. However, it can be seen that the size of $\mathcal{A}$ grows exponentially with A. In addition, A also grows as the number of users K increases. Thus, searching over the entire list $\mathcal{A}$ can be prohibitively complex when the number of users is large.

On the other hand, the proposed NN search method finds a set of M symbol vectors in $\mathcal{A}$ that are nearest to $\tilde{\mathbf{x}}$, then searches over that smaller set for the final solution. In this way, the NN search method can limit the computational complexity. Note that a symbol vector in this context is any element of $\mathcal{A}$. Let $\mathcal{X}_M = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M\}$ denote the set of the M nearest symbol vectors to $\tilde{\mathbf{x}}$. The larger M is, the higher the probability that the set $\mathcal{X}_M$ contains the true symbol vector. However, a large value of M will result in more computation for the search. Therefore, M should be chosen to achieve a good trade-off between detection accuracy and computational complexity. The value of M can be chosen by empirical evaluations. The main challenge here is how to find the M nearest symbol vectors to $\tilde{\mathbf{x}}$ quickly and efficiently. To address this problem, we employ the following notation and definitions.

For any two symbol vectors $\mathbf{x} \in \mathcal{A}$ and $\mathbf{x}' \in \mathcal{A}$, let $d(\mathbf{x}, \mathbf{x}')$ denote the number of position indices at which the elements of $\mathbf{x}$ are different from the corresponding elements of $\mathbf{x}'$. Since each element of $\mathbf{x}$ and $\mathbf{x}'$ belongs to a finite set of just one or two elements, $d(\mathbf{x}, \mathbf{x}')$ is actually the Hamming distance between $\mathbf{x}$ and $\mathbf{x}'$.

Definition 5.1 (Neighbor of a symbol vector)**.** *A symbol vector* $\mathbf{x}$ *is called a neighbor of another symbol vector* $\mathbf{x}'$*, or vice versa, when the Hamming distance between them is one, i.e.,* $d(\mathbf{x}, \mathbf{x}') = 1$*.*

Definition 5.2 (Neighbor of a set)**.** *Given a set of symbol vectors* $\mathcal{S}$ *and another symbol vector* $\mathbf{x} \notin \mathcal{S}$*, let*

$$d_{\min}(\mathbf{x}, \mathcal{S}) = \min_{\mathbf{x}' \in \mathcal{S}} d(\mathbf{x}, \mathbf{x}'). \tag{5.34}$$

The symbol vector $\mathbf{x}$ *is called a neighbor of* $\mathcal{S}$ *if and only if* $d_{\min}(\mathbf{x}, \mathcal{S}) = 1$*, or in other words, if and only if* $\mathbf{x}$ *is the neighbor of at least one member of* $\mathcal{S}$*.*

Let $\mathcal{N}(\mathbf{x})$ and $\mathcal{N}(\mathcal{S})$ denote the set of neighbors of symbol vector $\mathbf{x}$ and set $\mathcal{S}$, respectively. Let $\mathcal{X}_M = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M\}$ with $\mathbf{x}_m \in \mathcal{A}$ and $m \in \{1, 2, \ldots, M\}$ denote the set of the M

nearest symbol vectors to $\tilde{\mathbf{x}}$ satisfying

$$\|\mathbf{x}_1 - \tilde{\mathbf{x}}\|^2 < \|\mathbf{x}_2 - \tilde{\mathbf{x}}\|^2 < \ldots < \|\mathbf{x}_M - \tilde{\mathbf{x}}\|^2 < \|\mathbf{x}_{\text{out}} - \tilde{\mathbf{x}}\|^2 \tag{5.35}$$

where $\mathbf{x}_{\text{out}}$ is any symbol vector in $\mathcal{A}$, but not in $\mathcal{X}_M$. Hence, $\mathbf{x}_m$ is the m^{th} nearest symbol vector to $\tilde{\mathbf{x}}$. Clearly, the nearest symbol vector $\mathbf{x}_1$ is obtained by applying symbol-by-symbol detection to $\tilde{\mathbf{x}}$. The problem now is how to efficiently find $\mathbf{x}_2, \ldots, \mathbf{x}_M$. The following proposition can be exploited to solve this problem.

Proposition 5.1. *The m^{th} nearest symbol vector $\mathbf{x}_m$ must be a neighbor of the set $\mathcal{X}_{m-1} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m-1}\}$, i.e.,*

$$\mathbf{x}_m \in \mathcal{N}(\mathcal{X}_{m-1}).$$

Proof. Please refer to Appendix E □

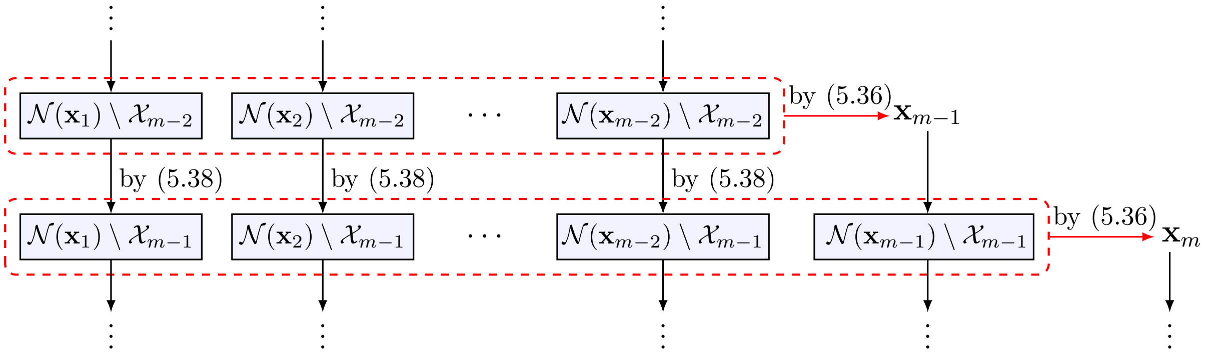
Proposition 5.1 indicates that we can find the m^{th} nearest symbol vector $\mathbf{x}_m$ from the neighbor set of $\mathcal{X}_{m-1}$, i.e.,

$$\mathbf{x}_m = \underset{\mathbf{x} \in \mathcal{N}(\mathcal{X}_{m-1})}{\arg\min} \|\mathbf{x} - \tilde{\mathbf{x}}\|^2 \tag{5.36}$$

where $\mathcal{N}(\mathcal{X}_{m-1})$ is the neighbor set of $\mathcal{X}_{m-1}$ and is given as

$$\begin{aligned} \mathcal{N}(\mathcal{X}_{m-1}) &= \left(\bigcup_{p=1}^{m-1} \mathcal{N}(\mathbf{x}_p) \right) \setminus \mathcal{X}_{m-1} \\ &= \bigcup_{p=1}^{m-1} \Big(\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-1} \Big). \end{aligned} \tag{5.37}$$

Hence, in order to find $\mathbf{x}_m$, we need to accomplish two tasks: (i) find $m-1$ subsets $\{\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-1}\}_{p=1,\ldots,m-1}$ and (ii) search for $\mathbf{x}_m$ within the subsets. The method of directly finding the $m-1$ subsets and then searching them for $\mathbf{x}_m$ is not efficient. In the following, we present a recursive strategy to obtain $\mathbf{x}_m$ quickly and efficiently.



$\mathcal{N}(\mathbf{x}_1) \setminus \mathcal{X}_{m-2}$
$\mathcal{N}(\mathbf{x}_2) \setminus \mathcal{X}_{m-2}$
$\cdots$
$\mathcal{N}(\mathbf{x}_{m-2}) \setminus \mathcal{X}_{m-2}$
by (5.36)
$\mathbf{x}_{m-1}$
by (5.38)
by (5.38)
by (5.38)
$\mathcal{N}(\mathbf{x}_1) \setminus \mathcal{X}_{m-1}$
$\mathcal{N}(\mathbf{x}_2) \setminus \mathcal{X}_{m-1}$
$\cdots$
$\mathcal{N}(\mathbf{x}_{m-2}) \setminus \mathcal{X}_{m-1}$
$\mathcal{N}(\mathbf{x}_{m-1}) \setminus \mathcal{X}_{m-1}$
by (5.36)
$\mathbf{x}_m$

Figure 5.8: Flowchart of the proposed nearest-neighbor search method. A recursive formation of sets is exploited to reduce the computational complexity. A subset $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-1}$ with $p \in \{1, \ldots, m-2\}$ is obtained by removing $\mathbf{x}_{m-1}$ from the subset $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}$ as given in (5.38). The last subset $\mathcal{N}(\mathbf{x}_{m-1}) \setminus \mathcal{X}_{m-1}$ is obtained by using $\mathbf{x}_{m-1}$ and other nearest symbol vectors. The m^{th} nearest symbol vector $\mathbf{x}_m$ is then obtained by searching over the $m-1$ subsets.

Note that the inner term on the right-hand side of (5.37) can be written as follows:

$$\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-1} = \Big(\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}\Big) \setminus \{\mathbf{x}_{m-1}\}. \tag{5.38}$$

Therefore, we can exploit (5.38) to obtain the first $m-2$ subsets $\{\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-1}\}_{p=1,\ldots,m-2}$ by removing $\mathbf{x}_{m-1}$ from $m-2$ other subsets $\{\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}\}_{p=1,\ldots,m-2}$, which were already obtained previously when we found $\mathbf{x}_{m-1}$. The last subset $\mathcal{N}(\mathbf{x}_{m-1}) \setminus \mathcal{X}_{m-1}$ is obtained by using $\mathbf{x}_{m-1}$ and the other nearest symbol vectors. A flowchart illustrating this recursive strategy is given in Fig. 5.8.

Remark 1: If the elements of $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}$ are already sorted in ascending order of distance to $\tilde{\mathbf{x}}$, then $\mathbf{x}_{m-1}$ can be removed from $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}$ by simply checking the first element of $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}$. The reason for this is that $\mathbf{x}_{m-1}$ is the $(m-1)^{\text{th}}$ nearest symbol vector, which means the distance from $\mathbf{x}_{m-1}$ to $\tilde{\mathbf{x}}$ cannot be greater than the distance from any element of $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}$ to $\tilde{\mathbf{x}}$. In addition, the elements of $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}$ are distinct and already sorted, and so if $\mathbf{x}_{m-1}$ exists in $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}$, it must be the first element of $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-2}$.

Remark 2: If the elements of each subset $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-1}$ are already sorted in ascending order of distance to $\tilde{\mathbf{x}}$, then the search over the $m-1$ subsets for $\mathbf{x}_m$ can be done by simply

Algorithm 4: Proposed Nearest-Neighbor Search.

```
    Input: $\tilde{\mathbf{x}}$, $\gamma$, $M$.
    Output: $\hat{\mathbf{x}}$.
 1  Find $\boldsymbol{\vartheta}$ and $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_{2U}$ based on $\boldsymbol{\vartheta}$;
 2  Let $|\mathcal{A}| = \prod_{i=1}^{2U} |\mathcal{A}_i|$;
 3  if $|\mathcal{A}| \leq M$ then
 4      Let $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \ldots \times \mathcal{A}_{2U}$;
 5      $\hat{\mathbf{x}} = \arg\min_{\mathbf{x} \in \mathcal{A}} \mathcal{P}(\mathbf{x})$;
 6  else
 7      Find $\mathbf{x}_1$ via symbol-by-symbol detection;
 8      Let $\mathcal{C}_1 = \mathrm{sort}\,(\mathcal{N}(\mathbf{x}_1))$;
 9      for $m = 2$ to $M$ do
10          Let $\mathcal{S}_m = \{\mathcal{C}_1[1], \mathcal{C}_2[1], \ldots, \mathcal{C}_{m-1}[1]\}$;
11          $\mathbf{x}_m = \arg\min_{\mathbf{x} \in \mathcal{S}_m} \|\mathbf{x} - \tilde{\mathbf{x}}\|^2$;
12          if $m < M$ then
13              for $p = 1$ to $m - 1$ do
14                  if $\mathcal{C}_p[1] = \mathbf{x}_m$ then
15                      Remove $\mathcal{C}_p[1]$ from $\mathcal{C}_p$;
16                  end
17              end
18              Let $\mathcal{C}_m = \mathrm{sort}\,(\mathcal{N}\,(\mathbf{x}_m))$;
19              for $p = 1$ to $m - 1$ do
20                  if $\mathcal{C}_m[1] = \mathbf{x}_p$ then
21                      Remove $\mathcal{C}_m[1]$ from $\mathcal{C}_m$;
22                  end
23              end
24          end
25      end
26      $\hat{\mathbf{x}} = \arg\min_{\mathbf{x} \in \mathcal{X}_M} \mathcal{P}(\mathbf{x})$;
27  end
28  return $\hat{\mathbf{x}}$;
```

searching over a list of $m-1$ candidates, where each candidate is the first element of a subset $\mathcal{N}(\mathbf{x}_p) \setminus \mathcal{X}_{m-1}$.

Based on the observations in Remarks 1 and 2, we propose the nearest-neighbor search method described in Algorithm 4. The key idea is to use the recursive strategy depicted in Fig. 5.8 and to implement the observations made in Remarks 1 and 2. Whenever forming a set $\mathcal{N}(\mathbf{x}_m)$, we sort its elements in ascending order of distance to $\tilde{\mathbf{x}}$ as described in lines 8 and 18 of Algorithm 4. In this way, we only need to sort $M-1$ times, and the remainder of the proposed algorithm only involves comparisons based on checking the first elements of the subsets. We denote $\mathcal{C}_1, \ldots, \mathcal{C}_{M-1}$ as the subsets corresponding to $\mathbf{x}_1, \ldots, \mathbf{x}_{M-1}$, respectively, and $\mathcal{C}_m[1]$ denotes the first element of the subset $\mathcal{C}_m$. Lines 10 and 11 implement Remark 2 to obtain $\mathbf{x}_m$. Remark 1 is implemented in lines 13-17. The last subset is obtained in lines 18-23. Finally, line 26 gives the final solution by searching for the highest-likelihood symbol vector among the M nearest symbol vectors.

5.6 Computational Complexity Analysis and Numerical Results

5.6.1 Computational Complexity Analysis

Here we present a Big-$\mathcal{O}$ computational complexity analysis for the considered channel estimation and data detection methods. The presented complexities only account for the online processing phase. Offline computations are excluded. Table 5.1 compares the complexity of different channel estimation methods. It can be seen that the complexity of BMMSE is the lowest and highest when the channels are i.i.d. and correlated, respectively. This is because the BMMSE estimation matrix can be computed offline for i.i.d. channels, but online for correlated channels. The complexity of BWZF is higher than that of the SVM method and the proposed FBM-CENet because the BWZF estimation matrix must also be computed online as it depends on the received signal. The complexity order of the proposed FBM-CENet is higher than BMMSE with i.i.d. channels, but scales more slowly than the SVM method since the SVM complexity is a super-linear function of T_t.

The complexity comparison of different data detection methods is given Table 5.2. Note that while the detection methods SVM and FBM-DetNet do not require preprocessing, BMMSE,

Table 5.1: Computational complexity comparison of channel estimation methods.

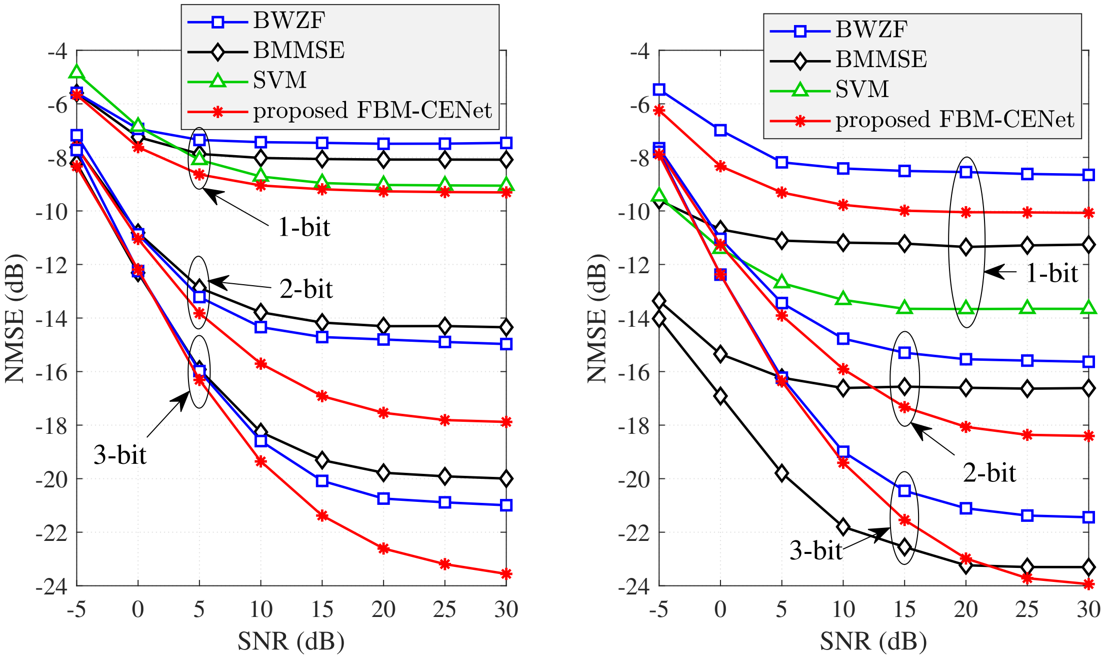
Method	**Complexity**
BMMSE	i.i.d. channels: $\mathcal{O}(UN^2T_{\mathrm{t}})$
	correlated channels: $\mathcal{O}(UN^3T_{\mathrm{t}}^2)$
BWZF	$\mathcal{O}\big(U^2N^3T_{\mathrm{t}}\big)$
Proposed SVM-based	$\mathcal{O}\big(UNT_{\mathrm{t}}f_{\mathtt{sl}}(T_{\mathrm{t}})\big)$
Proposed FBM-CENet	$\mathcal{O}\big(UN^2LT_{\mathrm{t}}\big)$

Table 5.2: Computational complexity comparison of data detection methods.

Method	**Preprocessing**	**Detection Stage**
BMMSE	$\mathcal{O}(N^3)$	$\mathcal{O}(UNT_{\mathrm{d}})$
BWZF	$\mathcal{O}(UN)$	$\mathcal{O}\big(U^2NT_{\mathrm{d}}\big)$
Proposed SVM-based	–	$\mathcal{O}\big(UNf_{\mathtt{sl}}(N)T_{\mathrm{d}}\big)$
Proposed B-DetNet	$\mathcal{O}(N^3)$	$\mathcal{O}\big(UNLT_{\mathrm{d}}\big)$
Proposed FBM-DetNet	–	$\mathcal{O}\big(UNLT_{\mathrm{d}}\big)$

BWZF, and B-DetNet require a preprocessing step due to the linearization process of the Bussgang decomposition. In the detection stage, BMMSE has the lowest complexity since it requires only one matrix-vector multiplication for each time slot. The complexity of BWZF is higher than BMMSE since the demultiplexing matrix of BWZF has to be re-computed in each time slot. The detection complexities of the proposed B-DetNet and FBM-DetNet are higher than the complexity of BMMSE, but lower than that of the SVM-based method.

The computational complexity of the proposed NN search method is $\mathcal{O}(MU\max\{M,N\}T_{\mathrm{d}})$ in the worst case. This complexity is mainly due to the detection step for $\hat{\mathbf{x}}$ and the **for** loops as described in Algorithm 4. The complexity of the full $\mathcal{A}$-space search method is $\mathcal{O}(|\mathcal{A}|UNT_{\mathrm{d}})$ where $|\mathcal{A}|$ can grow exponentially with U.



(a) Uncorrelated NLoS channels.

(b) Spatially correlated and mixed LoS-NLoS channels.

Figure 5.9: Channel estimation performance comparison for a given pilot matrix with $U = 4$, $L = 8$, and $N = 32$.

5.6.2 Numerical Results

Simulation Setting

Here we present numerical results that illustrate the superior performance of the proposed channel estimation and data detection networks. For training the networks, we use TensorFlow [108] and the Adam optimizer [109] with a learning rate that starts at 0.002 and decays at a rate of 0.97 after every 100 training epochs. The size of each training batch is set to 1000. The input of the first layer is set to a zero vector. When the pilot matrix $\mathbf{P}$ is trainable, we use the soft quantization model in (5.14) and (5.15) for the training phase and set $c_1 = 0.01$ and $c_2 = c_3 = c_4 = 1000$. For the channel estimation phase, we set the training length to be five times the number of users, i.e., $T_\mathrm{t} = 5U$.

We consider the following channel model:

$$\mathbf{H}^{\mathbb{C}} = \mathbf{H}^{\mathbb{C},\texttt{LoS}}\boldsymbol{\Xi}^{\texttt{LoS}} + \mathbf{H}^{\mathbb{C},\texttt{NLoS}}\boldsymbol{\Xi}^{\texttt{NLoS}}, \tag{5.39}$$

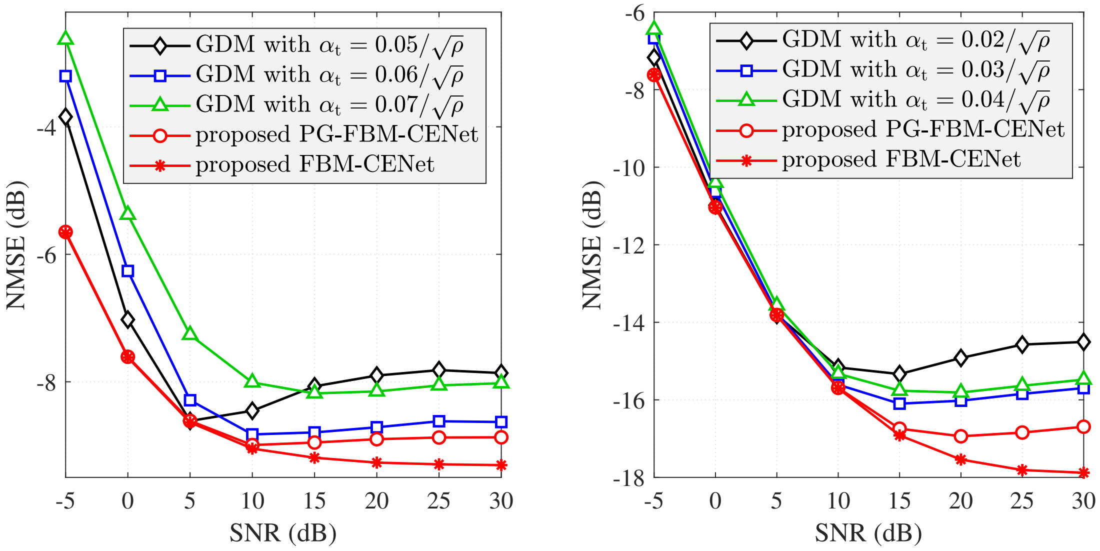
where $\mathbf{H}^{\mathbb{C},\texttt{LoS}}\boldsymbol{\Xi}^{\texttt{LoS}}$ and $\mathbf{H}^{\mathbb{C},\texttt{NLoS}}\boldsymbol{\Xi}^{\texttt{NLoS}}$ account for the Line-of-Sight (LoS) and Non-Line-of-Sight (NLoS) channels, respectively. The matrices $\boldsymbol{\Xi}^{\texttt{LoS}}$ and $\boldsymbol{\Xi}^{\texttt{NLoS}}$ are diagonal and defined as $\boldsymbol{\Xi}^{\texttt{LoS}} = \mathrm{diag}(\xi_1^{\texttt{LoS}}, \ldots, \xi_U^{\texttt{LoS}})$ and $\boldsymbol{\Xi}^{\texttt{NLoS}} = \mathrm{diag}(\xi_1^{\texttt{NLoS}}, \ldots, \xi_U^{\texttt{NLoS}})$ where

$$\xi_k^{\texttt{LoS}} = \sqrt{\frac{\kappa_u}{\kappa_u + 1}} \text{ and } \xi_u^{\texttt{NLoS}} = \sqrt{\frac{1}{\kappa_u + 1}}, \tag{5.40}$$

and κ_u is the Rician factor of the channel from the u-th user to the BS. If $\kappa_u = 0$, there is no LoS component between user-u and the BS. Let $\mathbf{H}^{\mathbb{C},\texttt{LoS}} = [\mathbf{h}_1^{\mathbb{C},\texttt{LoS}}, \ldots, \mathbf{h}_U^{\mathbb{C},\texttt{LoS}}]$ and $\mathbf{H}^{\mathbb{C},\texttt{NLoS}} = [\mathbf{h}_1^{\mathbb{C},\texttt{NLoS}}, \ldots, \mathbf{h}_U^{\mathbb{C},\texttt{NLoS}}]$. The LoS channel vector $\mathbf{h}_u^{\mathbb{C},\texttt{LoS}}$ is given as [110]

$$\mathbf{h}_u^{\mathbb{C},\texttt{LoS}} = \sqrt{\gamma_u}e^{j\varphi_u}[1, e^{j2\pi d_{\texttt{A}}\sin(\theta_u)}, \ldots, e^{j2\pi d_{\texttt{A}}(N-1)\sin(\theta_u)}]^T \tag{5.41}$$

where γ_u is the large-scale fading coefficient, $\varphi_u \in [0, 2\pi]$ is a random phase shift, and $d_{\texttt{A}}$ is the antenna spacing parameter (in fractions of a wavelength), and $-\pi/3 \leq \theta_u \leq \pi/3$ is the angle-of-arrival seen at the BS for user-u. The NLoS channel is given as $\mathbf{h}_u^{\mathbb{C},\texttt{NLoS}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}_u^{\mathbb{C}})$ where $\mathrm{tr}(\mathbf{C}_u^{\mathbb{C}})/N = \gamma_u$. Note that the channels between a user and different receive antennas can be correlated, but the channels between the users and the BS are uncorrelated. The large-scale fading coefficient is modeled (in dB) as [111] $\gamma_u = -30.18 - 26\log_{10}(d_u) + F_u$ where d_u is the distance between user-u and the BS, and $F_u \sim \mathcal{N}(0, \sigma_{\texttt{sh}}^2)$ is the shadow fading coefficient with $\sigma_{\texttt{sh}} = 4$. The Rician factor is given as $\kappa_u = 13 - 0.03d_u$ (dB) [111]. We assume perfect transmit power control at the users so that the received signal powers of different users are the same.



(a) 1-bit ADCs.

(b) 2-bit ADCs.

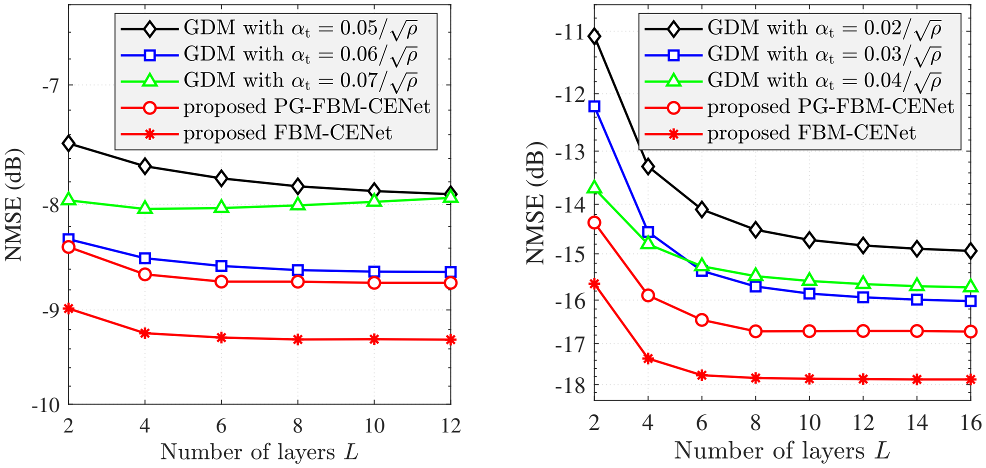
Figure 5.10: Channel estimation performance of GDM versus the proposed FBM-CENet with $U = 4$, $L = 8$, and $N = 32$.

Channel Estimation Performance Evaluation and Comparison

We compare the channel estimation performance of different methods in terms of normalized mean squared error (NMSE), defined here as $\mathrm{NMSE} = \mathbb{E}[\|\hat{\mathbf{H}} - \mathbf{H}^{\mathbb{C}}\|_{\mathrm{F}}^2]/\mathbb{E}[\|\mathbf{H}^{\mathbb{C}}\|_{\mathrm{F}}^2]$, where $\hat{\mathbf{H}}$ is an estimate of the channel $\mathbf{H}^{\mathbb{C}}$. When the pilot matrix is pre-specified, it is assumed to contain U columns of a $T_{\mathrm{t}} \times T_{\mathrm{t}}$ discrete Fourier transform (DFT) matrix. In particular, the u^{th} row of the pilot matrix $\mathbf{X}_{\mathrm{t}}^{\mathbb{C}}$ is column $(u+1)$ of the DFT matrix.

Fig. 5.9 presents a performance comparison of different channel estimation methods for the given DFT-based pilot matrix and considering both uncorrelated NLoS and spatially correlated mixed LoS-NLoS channels. Numerical results for the uncorrelated NLoS scenario are given in Fig. 5.9a where we set $\kappa_u = 0$ for all u, and $\mathbf{h}_u^{\mathbb{C},\mathtt{NLoS}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_N)$. It can be seen from Fig. 5.9a that the proposed FBM-CENet significantly outperforms existing methods. For the case of one-bit ADCs, it is observed that the proposed FBM-CENet slightly outperforms the SVM-based method at medium-to-high SNRs. However, at low SNRs, the performance gap between FBM-CENet and the SVM method is larger. For few-bit ADCs, it is clear that FBM-CENet significantly outperforms other existing channel estimation methods. Note that the SVM-based method was specifically designed for one-bit ADCs, and thus SVM results for other ADC resolutions are not available. The BWZF method does not perform well for one-bit ADCs since it gives a higher weight to signals with lower variance. However, for one-bit ADCs, there is only one bin on each side of the quantization threshold, and thus the weighting has no impact in this case. On the other hand, higher resolution ADCs result in more quantization bins and thus different weights come into play, and thus we see that BWZF performs better with few-bit quantization.

Numerical results for spatially correlated mixed LoS-NLoS channels are provided in Fig. 5.9b. For this scenario, we use the typical urban correlation model as in [33] and set $10 \leq d_u \leq 1000$. For the case of one-bit ADCs, the SVM-based method gives the best performance, while the proposed FBM-CENet performs worse than the SVM-based and BMMSE methods, but better than BWZF. The reason for this is because both BMMSE and SVM exploit knowledge of the channel correlation matrix, which is not used by FBM-CENet. Only received signals and pilot matrix are used by FBM-CENet. However, for few-bit ADCs, the proposed FBM-CENet approach outperforms both BMMSE and BWZF at higher SNRs even without using information about the channel statistics (recall that the SVM method only applies in the one-bit case). BMMSE gives the best performance at low SNRs for an additional reason beyond its use of the channel correlation information, namely that its use of the Bussgang decomposition is more accurate when the received signal is Gaussian, which becomes a better approximation as the power of the Gaussian noise increases. However, BMMSE uses an approximation for the Bussgang decomposition with few-bit ADCs that limits its performance at higher SNRs where FBM-CENet is superior. Note that BMMSE and SVM were implemented assuming perfect knowledge of the channel correlation, which may not be available in practice. FBM-CENet still provides very good performance without any relying on any correlation information, but extending the network to be able to exploit this information is an interesting area for future work.



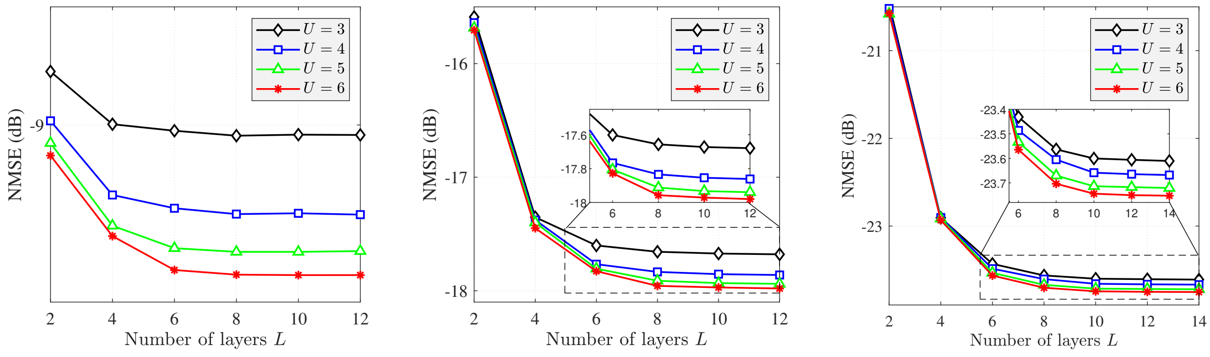
(a) 1-bit ADCs.

(b) 2-bit ADCs.

Figure 5.11: Channel estimation performance of GDM versus the proposed FBM-CENet with different values of L, $U = 4$, $N = 32$, and $\mathrm{SNR} = 30$ dB.

In the following comparisons and evaluations, from Fig. 5.10 to Fig. 5.13, we present results for uncorrelated NLoS channels since we found that the results were similar for spatially correlated mixed LoS-NLoS channels. Fig. 5.10 compares the proposed FBM-CENet with PG-FBM-CENet as well as the conventional gradient descent method (GDM) in (5.13) using a constant step size α_{t} for all iterations. The step size α_{t} used in GDM is normalized by the SNR as we found this gives stable performance for different SNR regimes. Note that FBM-CENet, PG-FBM-CENet, and GDM use the same number of layers (iterations) so that they have the same complexity. Simulation results show that the proposed FBM-CENet significantly outperforms PG-FBM-CENet and the conventional GDM. This results because, while GDM uses a common step size in all iterations and PG-FBM-CENet only optimizes the step sizes, the proposed FBM-CENet learns an optimal path by jointly optimizing the optimal step sizes $\{\alpha_{\mathrm{t}}^{(\ell)}\}$ as well as the optimal scaling parameter β_{t}.

In practice, the step size α_{t} can also be tuned by, for example, the backtracking line search method. However, this method requires an inner search loop in each iteration and therefore significantly increases the computational complexity compared to using fixed step sizes. Note that GDM, PG-FBM-CENet, and FBM-CENet presented above use fixed step sizes.



(a) 1-bit ADCs.

(b) 2-bit ADCs.

(c) 3-bit ADCs.

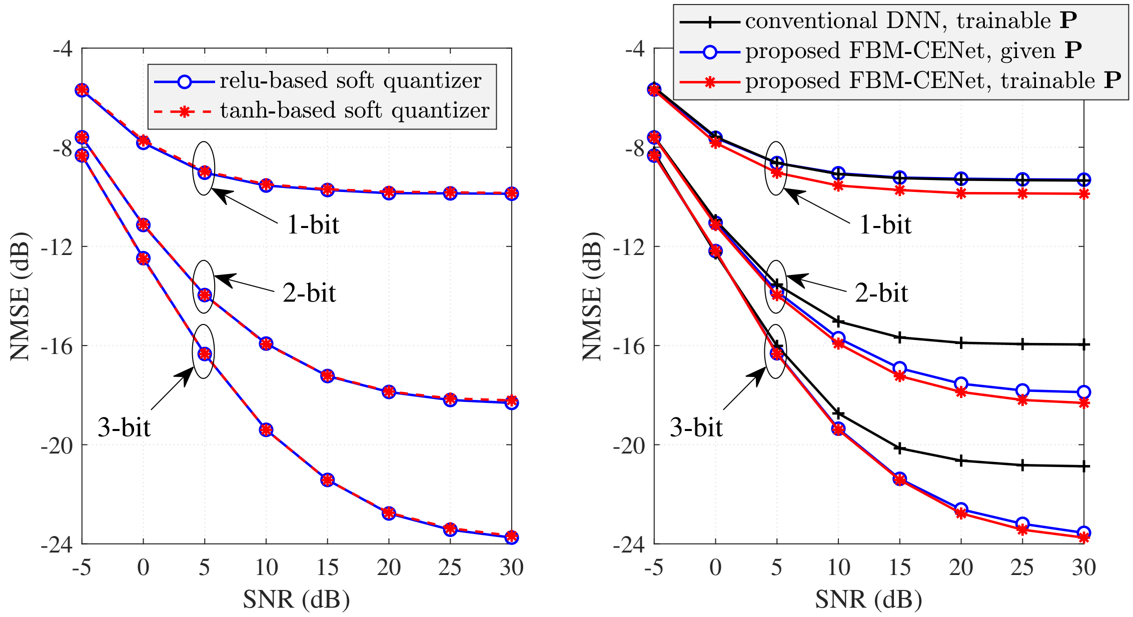
Figure 5.12: Channel estimation performance of the proposed FBM-CENet with various values of U and L, $N = 32$, and SNR $= 30$ dB.

For PG-FBM-CENet and FBM-CENet, the step sizes are obtained by the training process. Thus, GDM, PG-FBM-CENet, and FBM-CENet have significantly lower complexity. In addition, the inner search loop in each iteration requires the calculation of the objective function (5.10), which is undefined when the argument of the logarithm approaches zero. In our investigation, this issue occurs frequently. Note however that although the value of the objective function (5.10) can become undefined, its gradient (5.12) is robust against this issue.

In Fig. 5.11, we evaluate GDM, PG-FBM-CENet, and FBM-CENet for different numbers of layers L. It is observed that the proposed FBM-CENet performs better than both PG-FBM-CENet and GDM for different values of L and also requires fewer layers for convergence.

We investigate the performance of FBM-CENet as U, L, and b vary in Fig. 5.12. We see that for a given bit resolution b, the number of layers L need not be increased as the number of users U increases. However, as the bit resolution increases, improved performance can be achieved with an increased number of layers. Specifically, with one-bit ADCs, we can fix the number of layers to 6 as U increases from 3 to 6. However, as the bit resolution increases to 2 and 3, it is best to increase the number of layers to 8 and 10, respectively.

In Fig. 5.13, we consider the case where the pilot matrix is trained concurrently with the

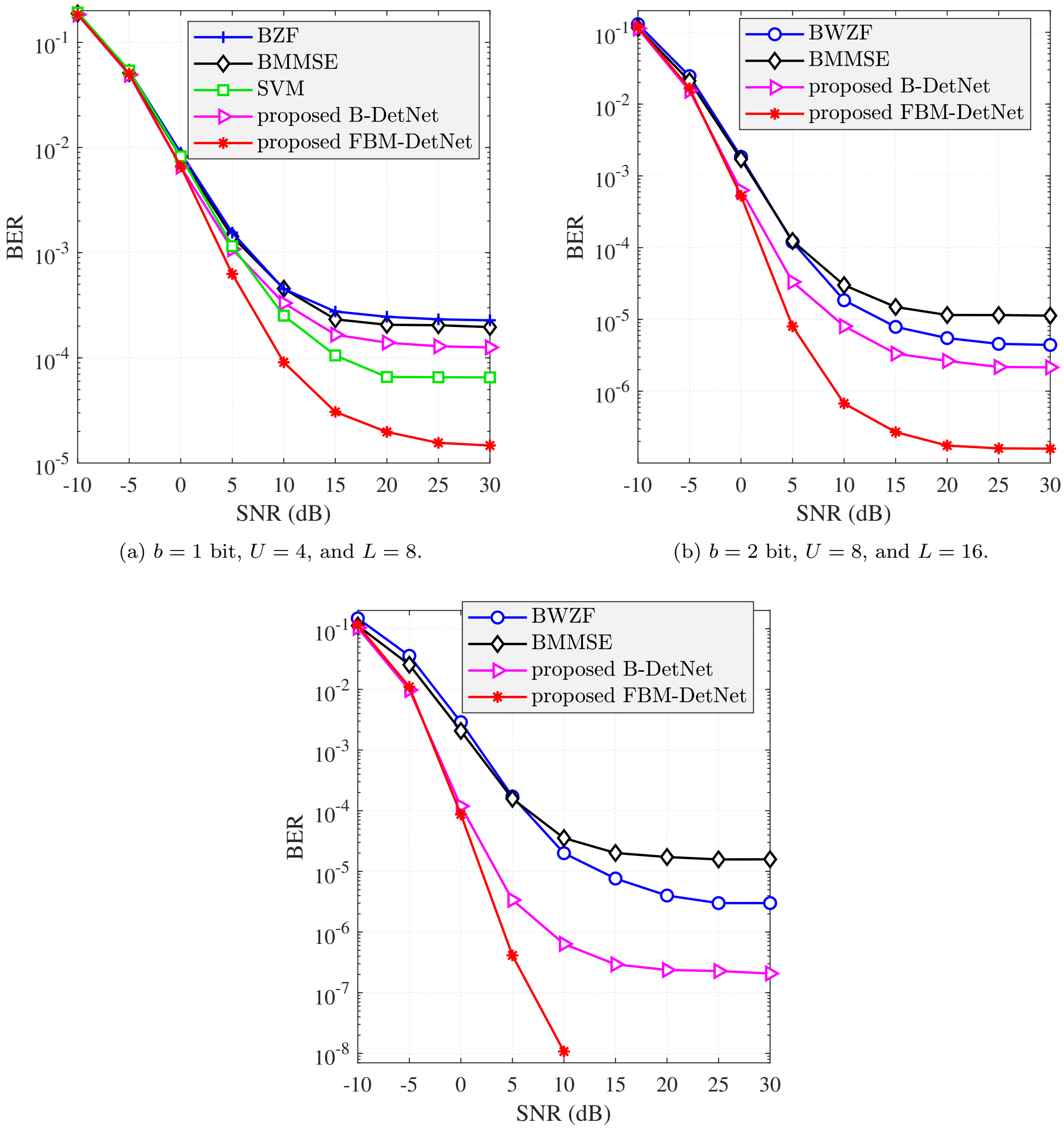


(a) Proposed FBM-CENet with relu-based and tanh-based soft quantizers.

(b) Proposed FBM-CENet versus conventional DNN.

Figure 5.13: Channel estimation performance comparison with trainable pilot matrix, $U = 4$, $L = 8$, and $N = 32$.

channel estimator. As mentioned earlier, when the pilot matrix is not given, we need to use a soft quantizer, based on either the ReLU or tanh function. In Fig. 5.13a, it is seen that the ReLU- and tanh-based soft quantizers give essientially identical performance. This is due to the fact that the parameters of the soft quantizers should be chosen so that they act similar to a hard quantizer. In Fig. 5.13b, the proposed FBM-CENet is compared with the existing conventional DNN-based method in [53] which also jointly optimizes the pilot matrix and the channel estimator. Note that we use the network structure and training method proposed in [53] to obtain the performance of the conventional DNN-based method. FBM-CENet significantly outperforms the channel estimator in [53] since the method of [53] uses the conventional data-driven DNN structure in Fig. 5.2a. On the other hand, the structure of FBM-CENet takes advantage of domain knowledge in the ML estimation framework. In Fig. 5.13b, we also include the channel estimation performance of FBM-CENet for a given orthogonal DFT-based pilot matrix in order to show that jointly optimizing the pilot matrix and the estimator can improve the estimation accuracy. This improvement is ob-



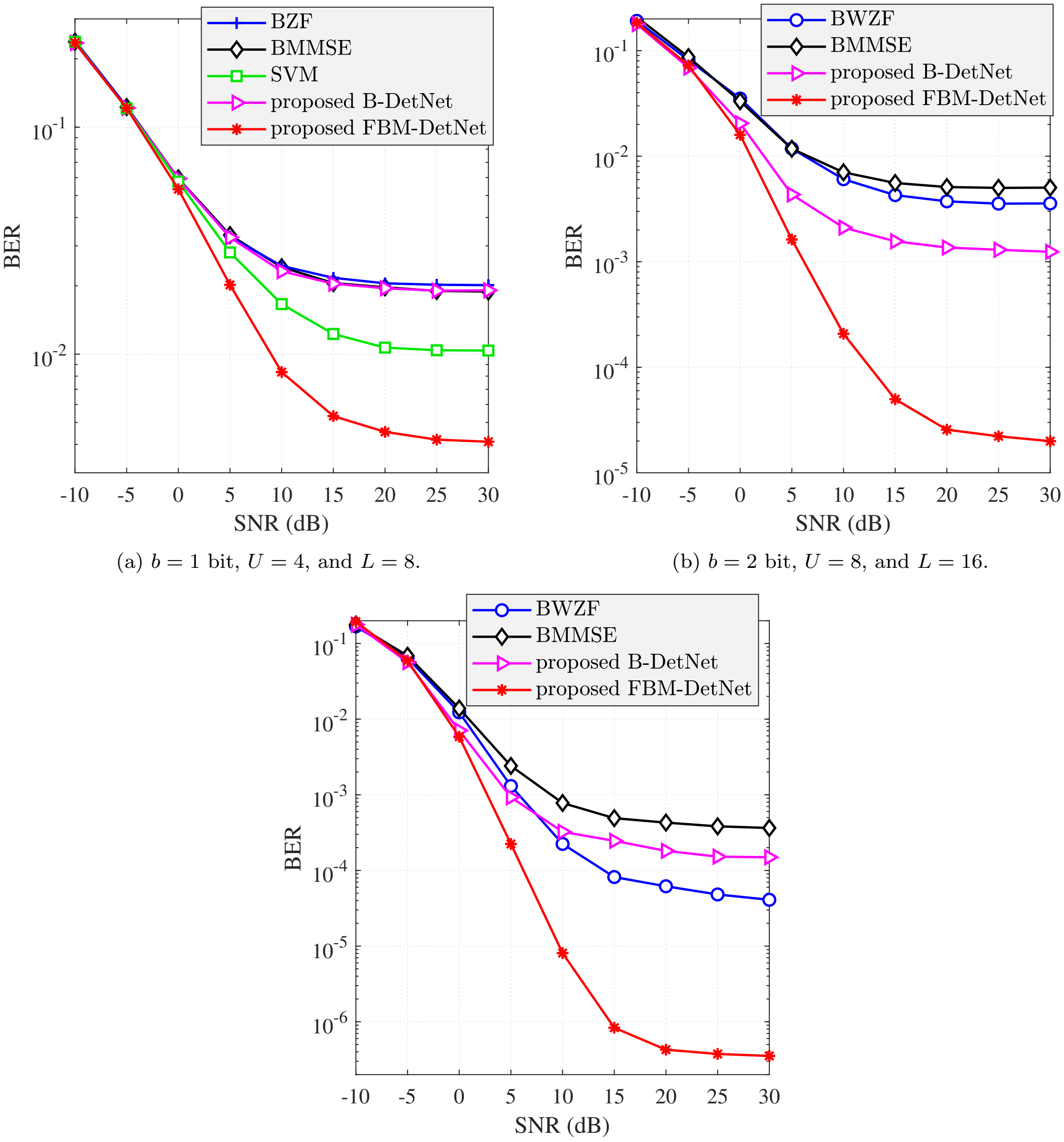
(a) $b = 1$ bit, $U = 4$, and $L = 8$.

(b) $b = 2$ bit, $U = 8$, and $L = 16$.

(c) $b = 3$ bit, $U = 16$, and $L = 24$.

Figure 5.14: Performance comparison for data detection methods with QPSK signalling and $N = 32$.

tained since orthogonal pilot data is known to be sub-optimal in low-resolution quantized systems [53]. When the pilot matrix is not given and treated as trainable, the training process of FBM-CENet produces a non-orthogonal pilot matrix that yields better performance than orthogonal pilots.



(a) $b = 1$ bit, $U = 4$, and $L = 8$.

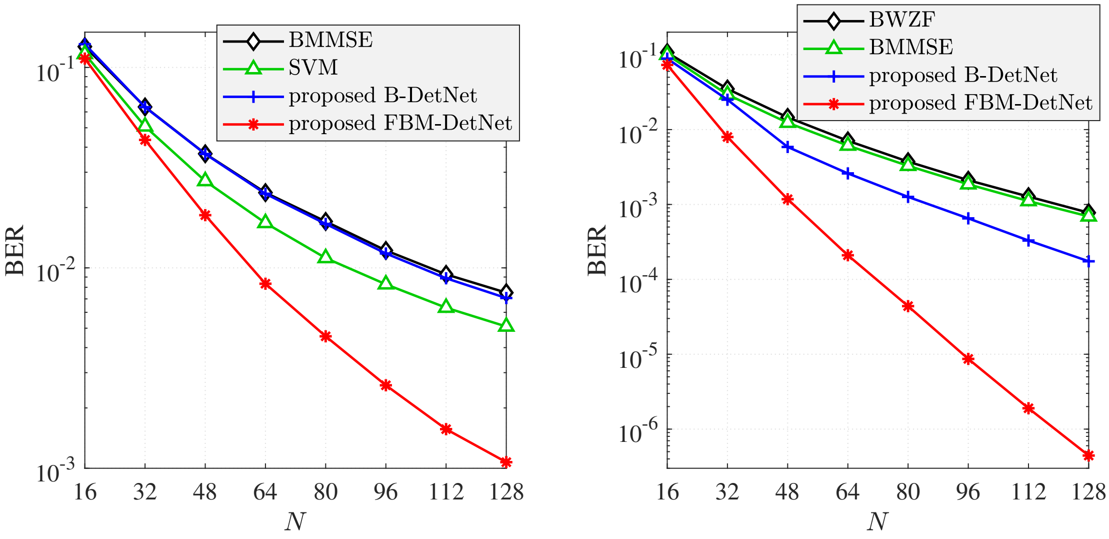
(b) $b = 2$ bit, $U = 8$, and $L = 16$.

(c) $b = 3$ bit, $U = 16$, and $L = 24$.

Figure 5.15: Performance comparison for data detection methods with 16-QAM signalling and $N = 64$.

Data Detection Performance Evaluation and Comparison

In the following, we present performance comparisons for data detection. Unless otherwise stated, uncorrelated NLoS channels are considered and the estimated CSI is obtained by FBM-CENet with a trainable pilot matrix. Comparisons given in Fig. 5.14 and Fig. 5.15 are for QPSK and 16-QAM signaling, respectively. The results show that FBM-DetNet



(a) 1-bit ADCs, $U = 4$, $L = 8$

(b) 2-bit ADCs, $U = 8$, $L = 16$

Figure 5.16: Data detection performance comparison for various values of N at 10-dB SNR and 16-QAM signalling.

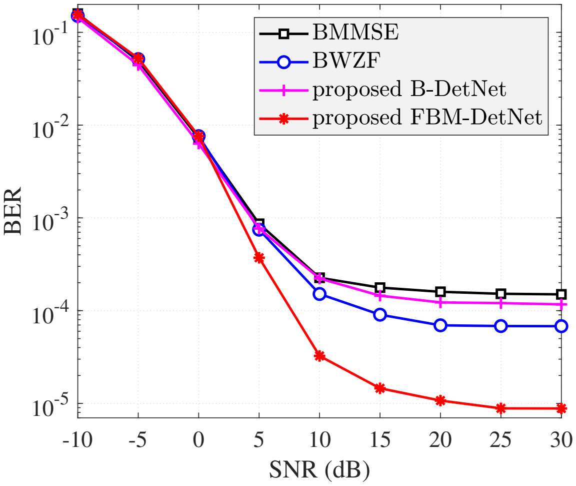


Figure 5.17: Data detection performance comparison with spatially correlated mixed LoS-NLoS channels, $b = 2$, $U = 4$, $N = 64$, $L = 8$, 16-QAM signalling, and BMMSE-based estimated CSI.

significantly outperforms other data detection methods. FBM-DetNet outperforms B-DetNet because FBM-DetNet is developed based on the original quantized system model whereas B-DetNet relies on the linearized system model in (5.21) whose effective noise $\mathbf{n}$ is approximated as Gaussian. Furthermore, the distortion covariance matrix $\mathbf{\Sigma_n}$ assumed by B-DetNet for the case of few-bit ADCs is approximate since a closed-form expression for $\mathbf{\Sigma_n}$ is intractable. For the case of 3-bit ADCs and 16-QAM signaling, B-DetNet performs worse than the BWZF method. As mentioned earlier, this is because BWZF performs better when there are more quantization bins (i.e., few-bit quantization), and also because B-DetNet is developed by unfolding the gradient descent of a linearized system, similar to the methodology applied in FS-Net [106] and DetNet [112], whose performance tends to degrade with higher dimensional constellations [113]. Note that FS-Net was developed for unquantized systems while B-DetNet is for the low-resolution quantized case. A good review of DNN-based detectors for unquantized systems can be found in [114].

In Fig. 5.16, we present a detection performance comparison for various values of N at 10-dB SNR and with 16-QAM signalling. It can be seen that the performance improvement of the proposed detection networks is maintained as the number of receive antennas increases. Since our derivations and methods assume no constraint on N, the proposed networks can work with an arbitrary number of receive antennas.

We provide a data detection performance comparison for spatially correlated mixed LoS-NLoS channels in Fig. 5.17 where the estimated CSI is obtained by the BMMSE method. It is still observed that the proposed FBM-DetNet gives the best performance. This shows that the proposed detection networks can work well with the estimated CSI given by not only FBM-CENet but also other channel estimation methods.

For the case of one-bit ADCs, the reformulated ML detection problem (5.29) reduces to the following form:

$$\hat{\mathbf{x}}_{\mathtt{ML}}^{\text{robust}} = \underset{\mathbf{x}^{\mathbb{C}} \in (\mathcal{M}^{\mathbb{C}})^U}{\arg\min} \sum_{i=1}^{2N} \log\left(1 + e^{-c\sqrt{2\rho}y_i \hat{\mathbf{h}}_i^T \mathbf{x}}\right). \tag{5.42}$$

The reformulated ML detection problem (5.42) does not share the non-robustness issue of (5.27), since if $\sqrt{2\rho}y_i\hat{\mathbf{h}}_i^T\mathbf{x}^\star$ is largely negative (due to $\text{sign}(\hat{\mathbf{h}}_i^T\mathbf{x}^\star) \neq y_i$ and large ρ), we have $\log(1+e^{-c\sqrt{2\rho}y_i\hat{\mathbf{h}}_i^T\mathbf{x}^\star}) \approx -c\sqrt{2\rho}y_i\hat{\mathbf{h}}_i^T\mathbf{x}^\star$. This approximation holds because $\log(1+e^t) \approx t$ for large t. Note that the value of $-c\sqrt{2\rho}y_i\hat{\mathbf{h}}_i^T\mathbf{x}^\star$ is finite for large ρ, and thus so is the objective function in (5.42) for all possible data vectors. Therefore, the reformulated ML

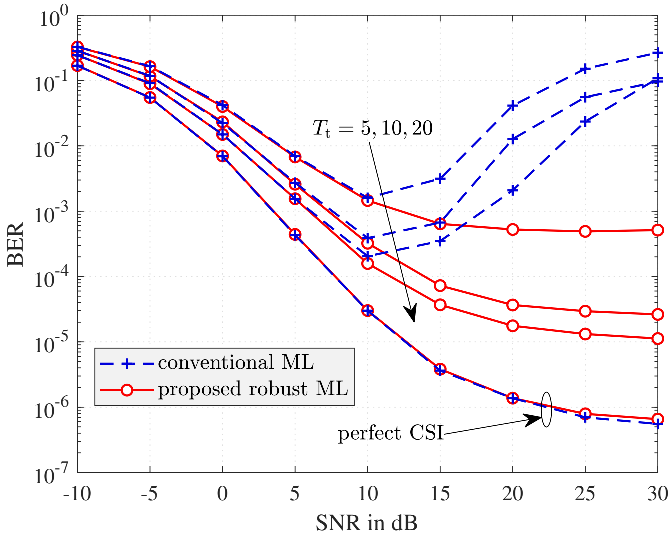
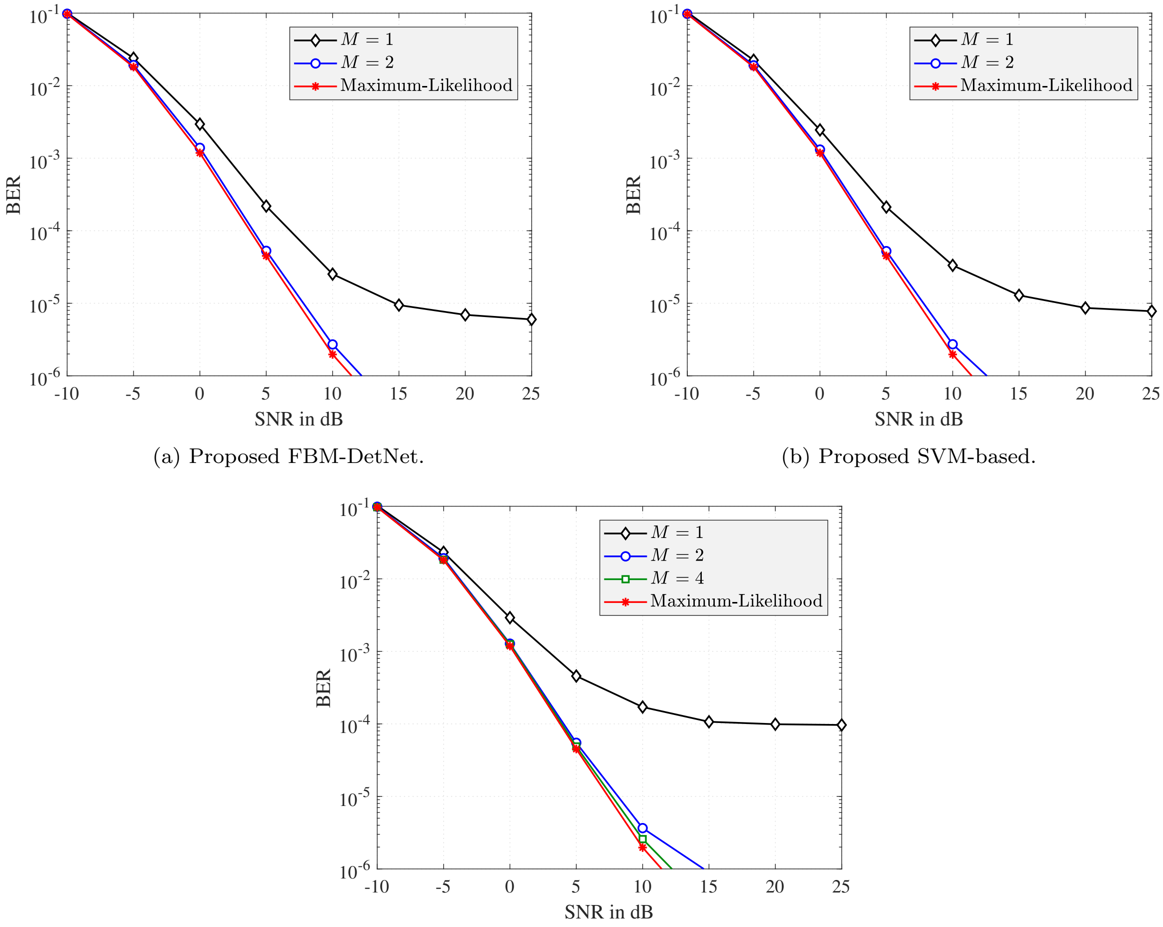


Figure 5.18: Performance comparison between the conventional and the proposed ML detection problems with $U = 2$, $N = 16$, and QPSK signaling. The BMMSE channel estimator is used with different training lengths T_t.

detection problem is more robust and (5.42) is more likely to yield $\mathbf{x}^\star$ as the optimal solution, unlike problem (5.27). Note that we have $\log(1 + e^t) \approx t$ for large t. However, a sequential computation by first evaluating e^t then the log function may result in an infinite value since e^t grows rapidly. Hence, one should use the approximation $\log(1 + e^t) \approx t$ when t is large, e.g., $t > 100$.

In Fig. 5.18, we verify the robustness of the reformulated ML detection problem (5.42) for the case of 1-bit ADCs when implemented with estimated CSI. We carried out simulations using the BMMSE channel estimator with different training lengths T_t. It can be seen from Fig. 5.18 that when the CSI is perfectly known, both the conventional and the proposed ML detection algorithms yield almost identical performance. However, when the CSI is imperfectly known, the performance of conventional ML detection is significantly degraded at high SNR, while the proposed robust ML detection algorithm remains stable.

For the NN search method, one-bit performance comparisons are given in Fig. 5.19 for the case of QPSK with $U = 4$ and $N = 32$, and Fig. 5.20 for the case of 16-QAM with $U = 8$ and $N = 128$. We set $\gamma = \frac{1}{2\sqrt{2}}$ for QPSK and $\gamma = \frac{1}{2\sqrt{10}}$ for 16-QAM. Here, we compare the BZF,



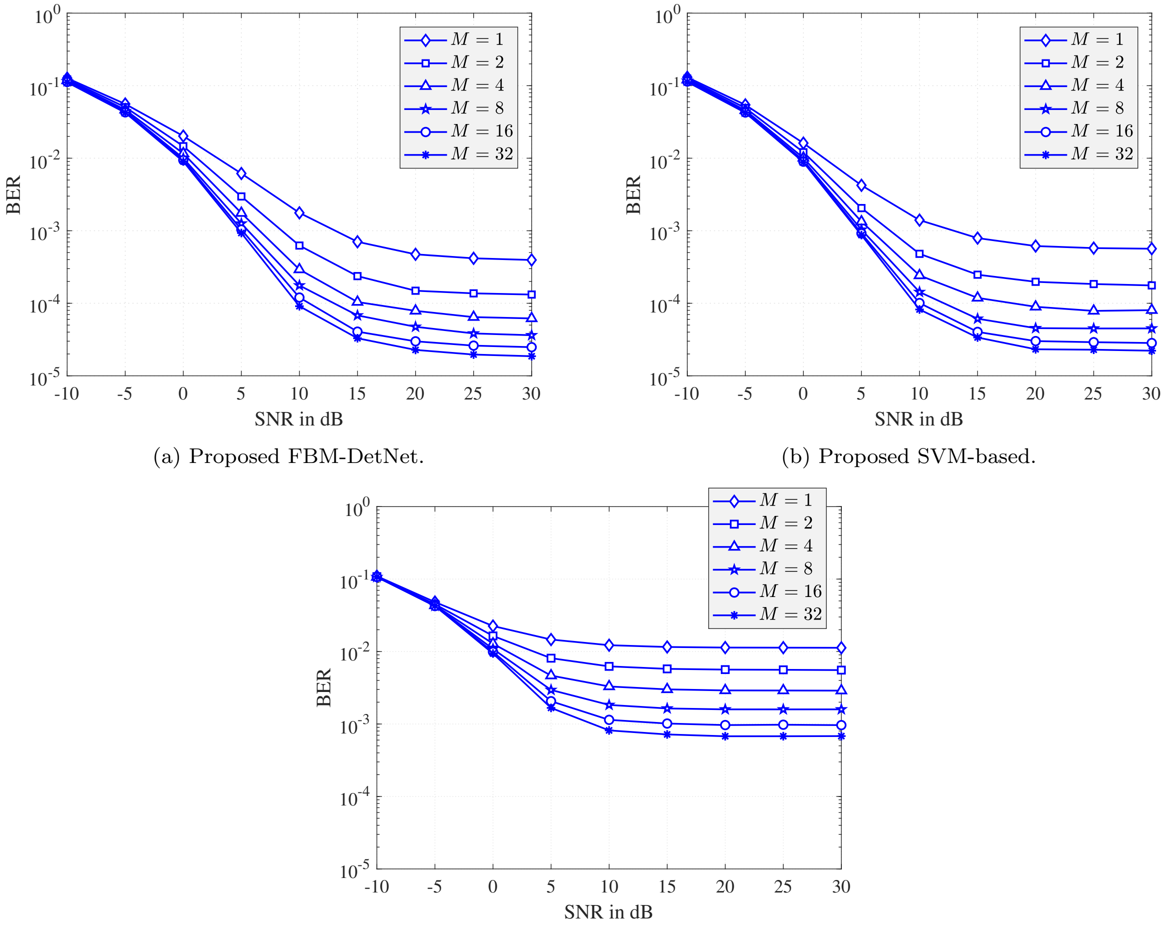
(a) Proposed FBM-DetNet.

(b) Proposed SVM-based.

(c) BZF.

Figure 5.19: Second stage performance comparison between different receivers with $b = 1$, $U = 4$, $N = 32$, QPSK signaling, and perfect CSI.

FBM-DetNet without the projector, and SVM-based receivers and omit BMMSE since the performance of BZF and BMMSE are comparable, and the complexity of BZF is lower than that of BMMSE. The case of $M = 1$ is equivalent to the use of symbol-by-symbol detection in the first stage. In this case, OBMNet provides the best performance, i.e., it yields the best initial detection results. When increasing M, the proposed NN search method in the second stage significantly improves the performance compared to the first stage. In Fig. 5.19, the BERs obtained with a small M, e.g., $M = 2$, are already close to the BER of the ML detection approach. The results in Fig. 5.20 clearly show that the performance can be improved by increasing M, but this requires more computation resources. Thus, one should



(a) Proposed FBM-DetNet.

(b) Proposed SVM-based.

(c) BZF.

Figure 5.20: Second stage performance comparison between different receivers with $b = 1$, $U = 8$, $N = 128$, 16-QAM signaling, and perfect CSI.

choose M to balance the detection accuracy and computational complexity. It should be noted that $|\mathcal{A}|$ is always a power of two, but M can be any positive integer number.

5.7 Conclusion

In this chapter, we have developed a channel estimation network (FBM-CENet) and two data detection networks (B-DetNet and FBM-DetNet) for massive MIMO systems with low-resolution ADCs. The proposed networks are model-driven and have special structures that can take advantage of domain-knowledge to efficiently address the severe non-linearity caused by the low-resolution ADCs. An interesting feature of the proposed FBM-CENet is that the pilot matrix directly plays the role of the weight matrices in the network structure, which makes it possible to jointly optimize the estimation network and the pilot signal by simply treating the pilot matrix as trainable parameters. The proposed detection networks are highly adaptive to the channel and easy to train since they have a small number of trainable parameters. Simulation results show that the proposed networks significantly outperform existing methods.

We have also proposed an NN search method to further improve the data detection performance. The proposed NN search method generates searches over a limited number of most likely candidates and thus helps contain the search complexity. A recursive strategy was proposed to obtain the set of nearest candidates efficiently and quickly so that the proposed NN search method can be implemented in an efficient manner.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

One practical solution for reducing hardware cost and power consumption in MIMO systems is to use low-resolution ADCs, due to their simple structure and very low power consumption. However, the severe nonlinearity of low-resolution ADCs causes significant distortions in the received signals and makes signal processing tasks such as channel estimation and data detection much more challenging compared to those in high-resolution systems. This dissertation exploits machine learning to develop low-complexity yet efficient and robust algorithms for channel estimation and data detection in MIMO systems with low-resolution ADCs. It has been shown that machine learning techniques such as K-means clustering, SVM, and DNN are powerful tools for addressing the channel estimation and data detection problems in MIMO systems with low-resolution ADCs.

Blind detection in MIMO systems with low-resolution ADCs is studied in Chapter 3 where two new learning methods for enhancing the performance were proposed. Numerical results demonstrate the performance improvement and robustness of the proposed learning methods over existing techniques. It was also observed that the two proposed learning methods require only a few iterations to converge. Chapter 3 also gives a performance analysis for the proposed learning methods in case of one-bit ADCs. Based on the analytical results, a new criterion for the transmit signal design problem has been proposed.

Chapter 4 showed that efficient and robust channel estimation and data detection in one-bit massive MIMO systems can be obtained through the SVM framework. SVM-based channel estimators for both uncorrelated and spatially correlated channels were developed. This chapter aslo proposed a two-stage SVM-based data detection method and an SVM-based joint CE-DD method. Finally, an extension of the proposed methods to OFDM systems with frequency-selective fading channels was derived. Simulation results revealed the superiority of the proposed SVM-based methods against existing ones and the gain is greatest for moderate to high SNR regimes.

In Chapter 5, model-driven deep neural networks for channel estimation, pilot signal design, and data detection were developed. The channel estimation network allows a joint optimization of the channel estimator and the pilot signal design. The detection networks are highly adaptive to the channel and easy to train since their structure contains a small number of trainable parameters. The developed networks were shown to have low complexities and outperform existing methods. Last but not least, Chapter 5 proposed an NN search method to further improve the data detection performance.

6.2 Future Work

In this section, we present interesting topics for future work.

1. *Variational Bayesian (VB) Approach for Low-Resolution MIMO Signal Processing:* VB inference is a powerful machine learning framework that provides approximation

of intractable posterior distributions. The VB inference framework can be used in lieu of traditional machine learning models when the channel is rapidly time-varying. Preliminary results have shown that VB inference is a promising approach for low-resolution MIMO signal processing [115].

2. *MIMO Channel Estimation and Data Detection with Spatial* $\Sigma\Delta$ *Quantization:* The idea of spatial $\Sigma\Delta$ quantization is to apply feedback and oversampling to the spatial domain [116–119], which provides noise shaping in space by placing antennas closer than half of the wavelength. Spatial $\Sigma\Delta$ structures have been leveraged for interference cancellation and beamforming [116, 120]. Early results in spectral efficiency analysis show that spatial $\Sigma\Delta$ quantization can significantly improve the system performance [121,122]. However, limited results have been reported for the MIMO channel estimation and data detection problems.

3. *Fully Low-Resolution MIMO Signal Processing:* Thus far, the dissertation has been considering low resolution at the ADCs. Low resolution can also be considered in baseband data processing. This will help significantly reduce circuit area, processing delay, and processing power consumption, which are critical in high speed data rate communication with low-cost devices. Preliminary results [123, 124] only consider simple linear receivers such MMSE. Learning-based signal processing framework using few-bit operations can offer signal performance advantage over existing linear processing methods.

Bibliography

[1] T. Barnett Jr., S. Jain, U. Andra, and T. Khurana. (2019, Mar.) Cisco visual networking index (VNI) global and americas/EMEAR mobile data traffic forecast, 2017–2022. [Online]. Available: https://www.cisco.com/c/dam/m/en_us/network-intelligence/service-provider/digital-transformation/knowledge-network-webinars/pdfs/190320-mobility-ckn.pdf

[2] D. Tse and P. Viswanath, *Fundamentals of wireless communication.* Cambridge university press, 2005.

[3] J. R. Hampton, *Introduction to MIMO communications.* Cambridge University Press, 2013.

[4] E. Björnson, J. Hoydis, and L. Sanguinetti, "Massive MIMO networks: Spectral, energy, and hardware efficiency," *Foundations and Trends in Signal Processing*, vol. 11, no. 3–4, pp. 154–655, 2017.

[5] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[6] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 74–80, Feb. 2014.

[7] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?" *IEEE J. Select. Areas in Commun.*, vol. 32, no. 6, pp. 1065–1082, June 2014.

[8] A. L. Swindlehurst, E. Ayanoglu, P. Heydari, and F. Capolino, "Millimeter-wave massive MIMO: The next wireless revolution?" *IEEE Commun. Mag.*, vol. 52, no. 9, pp. 56–62, Sept. 2014.

[9] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," *IEEE J. Select. Topics in Signal Process.*, vol. 8, no. 5, pp. 742–758, Oct. 2014.

[10] J. Hoydis, S. ten Brink, and M. Debbah, “Massive MIMO in the UL/DL of cellular networks: How many antennas do we need?” *IEEE J. Select. Areas in Commun.*, vol. 31, no. 2, pp. 160–171, Feb. 2013.

[11] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, “Energy and spectral efficiency of very large multiuser MIMO systems,” *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1436–1449, Apr. 2013.

[12] T. L. Marzetta, *Fundamentals of massive MIMO.* Cambridge University Press, 2016.

[13] E. Bj ornson, J. Hoydis, and L. Sanguinetti, “Massive MIMO has unlimited capacity,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 574–590, Jan. 2018.

[14] R. H. Walden, “Analog-to-digital converter survey and analysis,” *IEEE J. Select. Areas in Commun.*, vol. 17, no. 4, pp. 539–550, Apr. 1999.

[15] B. Murmann, “The race for the extra decibel: A brief review of current ADC performance trajectories,” *IEEE Solid-State Circuits Magazine*, vol. 7, no. 3, pp. 58–66, Summer 2015.

[16] *Common Public Radio Interface (CPRI); Interface Specification, CPRI Specification v6.0*, Ericsson AB, Huawei Technol., NEC Corp., Alcatel Lucent, and Nokia Siemens Netw., Aug. 2013.

[17] D. Hui and D. L. Neuhoff, “Asymptotic analysis of optimal fixed-rate uniform scalar quantization,” *IEEE Trans. Inform. Theory*, vol. 47, no. 3, pp. 957–977, Mar. 2001.

[18] N. Al-Dhahir and J. M. Cioffi, “On the uniform ADC bit precision and clip level computation for a Gaussian signal,” *IEEE Trans. Signal Process.*, vol. 44, no. 2, pp. 434–438, Feb. 1996.

[19] A. Mezghani and J. A. Nossek, “On ultra-wideband MIMO systems with 1-bit quantized outputs: Performance analysis and input optimization,” in *Proc. IEEE Int. Symp. Inf. Theory*, 2007, pp. 1286–1289.

[20] A. Mezghani and J. A. Nossek, “Capacity lower bound of MIMO channels with output quantization and correlated noise,” in *Proc. IEEE Int. Symp. Inf. Theory*, 2012.

[21] P. Dong, H. Zhang, W. Xu, G. Y. Li, and X. You, “Performance analysis of multiuser massive MIMO with spatially correlated channels using low-precision ADC,” *IEEE Commun. Letters*, vol. 22, no. 1, pp. 205–208, Jan. 2018.

[22] J. Mo and R. W. Heath, “High SNR capacity of millimeter wave MIMO systems with one-bit quantization,” in *Proc. Inf. Theory and Applications Workshop*, 2014.

[23] ——, “Capacity analysis of one-bit quantized MIMO systems with transmitter channel state information,” *IEEE Trans. Signal Process.*, vol. 63, no. 20, pp. 5498–5512, Oct. 2015.

[24] L. Fan, S. Jin, C. Wen, and H. Zhang, "Uplink achievable rate for massive MIMO systems with low-resolution ADC," *IEEE Commun. Letters*, vol. 19, no. 12, pp. 2186–2189, Dec. 2015.

[25] J. Mo, A. Alkhateeb, S. Abu-Surra, and R. W. Heath, "Hybrid architectures with few-bit ADC receivers: Achievable rates and energy-rate tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 16, no. 4, pp. 2274–2287, Apr. 2017.

[26] K. Roth and J. A. Nossek, "Achievable rate and energy efficiency of hybrid and digital beamforming receivers with low resolution ADC," *IEEE J. Select. Areas in Commun.*, vol. 35, no. 9, pp. 2056–2068, Sept. 2017.

[27] N. Liang and W. Zhang, "Mixed-ADC massive MIMO," *IEEE J. Select. Areas in Commun.*, vol. 34, no. 4, pp. 983–997, Apr. 2016.

[28] ——, "Mixed-ADC massive MIMO uplink in frequency-selective channels," *IEEE Trans. Commun.*, vol. 64, no. 11, pp. 4652–4666, Nov. 2016.

[29] C. Mollén, J. Choi, E. G. Larsson, and R. W. Heath, "Uplink performance of wideband massive MIMO with one-bit ADCs," *IEEE Trans. Wireless Commun.*, vol. 16, no. 1, pp. 87–100, Jan. 2017.

[30] S. Jacobsson, G. Durisi, M. Coldrey, U. Gustavsson, and C. Studer, "Throughput analysis of massive MIMO uplink with low-resolution ADCs," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 4038–4051, June 2017.

[31] J. Choi, J. Mo, and R. W. Heath, "Near maximum-likelihood detector and channel estimator for uplink multiuser massive MIMO systems with one-bit ADCs," *IEEE Trans. Commun.*, vol. 64, no. 5, pp. 2005–2018, May 2016.

[32] C. Risi, D. Persson, and E. G. Larsson, "Massive MIMO with 1-bit ADC," *arXiv:1404.7736 [cs.IT]*, 2014.

[33] Y. Li, C. Tao, G. Seco-Granados, A. Mezghani, A. L. Swindlehurst, and L. Liu, "Channel estimation and performance analysis of one-bit massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 4075–4089, Aug. 2017.

[34] S. Rao, A. L. Swindlehurst, and H. Pirzadeh, "Massive MIMO channel estimation with 1-bit spatial sigma-delta ADCs," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process.*, Brighton, United Kingdom, May 2019, pp. 4484–4488.

[35] Z. Shao, L. T. N. Landau, and R. C. d. Lamare, "Oversampling based channel estimation for 1-bit large-scale multiple-antenna systems," in *Proc. Int. ITG Workshop on Smart Antennas*, Vienna, Austria, April 2019.

[36] Z. Shao, L. T. N. Landau, and R. C. de Lamare, "Channel estimation using 1-bit quantization and oversampling for large-scale multiple-antenna systems," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process.*, Brighton, United Kingdom, May 2019, pp. 4669–4673.

[37] K. Gao, N. J. Estes, B. Hochwald, J. Chisum, and J. N. Laneman, “Power-performance analysis of a simple one-bit transceiver,” in *Proc. Information Theory and Applications Workshop*, San Diego, CA, USA, Feb. 2017.

[38] F. Liu, H. Zhu, C. Li, J. Li, P. Wang, and P. Orlik, “Angular-Domain channel estimation for one-bit massive MIMO systems: Performance bounds and algorithms,” *IEEE Trans. Veh. Technol. (Early Access)*, 2020.

[39] I. Kim, N. Lee, and J. Choi, “Dominant channel estimation via MIPS for large-scale antenna systems with one-bit ADCs,” in *Proc. IEEE Global Commun. Conf.*, Abu Dhabi, United Arab Emirates, Dec. 2018.

[40] H. Kim and J. Choi, “Channel AoA estimation for massive MIMO systems using one-bit ADCs,” *Journal of Communications and Networks*, vol. 20, no. 4, pp. 374–382, Aug. 2018.

[41] H. Kim and J. Choi, “Channel estimation for spatially/temporally correlated massive MIMO systems with one-bit ADCs,” *EURASIP J. Wireless Commun. and Networking*, vol. 2019, no. 1, p. 267, 2019.

[42] B. Srinivas, K. Mawatwal, D. Sen, and S. Chakrabarti, “An iterative semi-blind channel estimation scheme and uplink spectral efficiency of pilot contaminated one-bit massive MIMO systems,” *IEEE Tran. Veh. Technol.*, vol. 68, no. 8, pp. 7854–7868, Aug. 2019.

[43] A. Mezghani and A. L. Swindlehurst, “Blind estimation of sparse broadband massive MIMO channels with ideal and one-bit ADCs,” *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2972–2983, June 2018.

[44] I. S. Kim and J. Choi, “Channel estimation via gradient pursuit for mmWave massive MIMO systems with one-bit ADCs,” *EURASIP J. Wireless Commun. and Networking*, vol. 2019, no. 1, p. 289, 2019.

[45] J. Mo, P. Schniter, and R. W. Heath, “Channel estimation in broadband millimeter wave MIMO systems with few-bit ADCs,” *IEEE Trans. Signal Process.*, vol. 66, no. 5, pp. 1141–1154, Mar. 2018.

[46] J. Rodríguez-Fernández, N. González-Prelcic, and R. W. Heath, “Channel estimation in mixed hybrid-low resolution MIMO architectures for mmWave communication,” in *Proc. Asilomar Conf. Signals, Systems and Computers*, Pacific Grove, CA, USA, Nov. 2016, pp. 768–773.

[47] C. Rusu, R. Mendez-Rial, N. Gonzalez-Prelcic, and R. W. Heath, “Adaptive one-bit compressive sensing with application to low-precision receivers at mmWave,” in *Proc. IEEE Global Commun. Conf.*, San Diego, CA, USA, Dec. 2015.

[48] S. Rao, A. Mezghani, and A. L. Swindlehurst, “Channel estimation in one-bit massive MIMO systems: Angular versus unstructured models,” *IEEE J. Select. Topics in Signal Process.*, vol. 13, no. 5, pp. 1017–1031, Sep. 2019.

[49] E. Balevi and J. G. Andrews, "Two-stage learning for uplink channel estimation in one-bit massive MIMO," in *Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, USA, Nov. 2019, pp. 1764–1768.

[50] Y. Dong, H. Wang, and Y.-D. Yao, "Channel estimation for one-bit multiuser massive MIMO using conditional GAN," *IEEE Commun. Letters*, vol. 25, no. 3, pp. 854–858, Mar. 2021.

[51] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Deep learning for massive MIMO with 1-bit ADCs: When more antennas need fewer pilots," *IEEE Wireless Commun. Letters*, vol. 9, no. 8, pp. 1273–1277, Aug. 2020.

[52] N. Kolomvakis, T. Eriksson, M. Coldrey, and M. Viberg, "Quantized uplink massive MIMO systems with linear receivers," in *Proc. IEEE Int. Conf. Commun.*, Dublin, Ireland, June 2020.

[53] D. H. N. Nguyen, "Neural network-optimized channel estimator and training signal design for MIMO systems with few-bit ADCs," *IEEE Signal Process. Letters*, vol. 27, pp. 1370–1374, 2020.

[54] S. Gao, P. Dong, Z. Pan, and G. Y. Li, "Deep learning based channel estimation for massive MIMO with mixed-resolution ADCs," *IEEE Commun. Letters*, vol. 23, no. 11, pp. 1989–1993, Nov. 2019.

[55] J. Zicheng, G. Shen, L. Nan, P. Zhiwen, and Y. Xiaohu, "Deep learning-based channel estimation for massive-MIMO with mixed-resolution ADCs and low-resolution information utilization," *IEEE Access*, vol. 9, pp. 54 938–54 950, Apr. 2021.

[56] Y. Ding, S. Chiu, and B. D. Rao, "Bayesian channel estimation algorithms for massive MIMO systems with hybrid analog-digital processing and low-resolution ADCs," *IEEE J. Select. Topics in Signal Process.*, vol. 12, no. 3, pp. 499–513, June 2018.

[57] A. Kaushik, E. Vlachos, J. Thompson, and A. Perelli, "Efficient channel estimation in millimeter wave hybrid MIMO systems with low resolution ADCs," in *Proc. European Signal Processing Conference*, Rome, Italy, Sept. 2018, pp. 1825–1829.

[58] J. Choi, D. J. Love, D. R. Brown, and M. Boutin, "Quantized distributed reception for MIMO wireless systems using spatial multiplexing," *IEEE Trans. Signal Process.*, vol. 63, no. 13, pp. 3537–3548, July 2015.

[59] S. Wang, Y. Li, and J. Wang, "Convex optimization based multiuser detection for uplink large-scale MIMO under low-resolution quantization," in *Proc. IEEE Int. Conf. Commun.*, Sydney, NSW, Australia, June 2014, pp. 4789–4794.

[60] A. Mezghani, M. Khoufi, and J. A. Nossek, "Maximum likelihood detection for quantized MIMO systems," in *Proc. Int. ITG Workshop on Smart Antennas*, Vienna, Austria, Feb. 2008, pp. 278–284.

[61] Y. Jeon, N. Lee, S. Hong, and R. W. Heath, "One-bit sphere decoding for uplink massive MIMO systems with one-bit ADCs," *IEEE Trans. Wireless Commun.*, vol. 17, no. 7, pp. 4509–4521, July 2018.

[62] C. K. Wen, C. J. Wang, S. Jin, K. K. Wong, and P. Ting, "Bayes-optimal joint channel-and-data estimation for massive MIMO with low-precision ADCs," *IEEE Trans. Signal Process.*, vol. 64, no. 10, pp. 2541–2556, May 2016.

[63] S. S. Thoota and C. R. Murthy, "Variational Bayes' joint channel estimation and soft symbol decoding for uplink massive MIMO systems with low resolution ADCs," *IEEE Trans. Commun.*, vol. 69, no. 5, pp. 3467–3481, May 2021.

[64] A. S. Lan, M. Chiang, and C. Studer, "Linearized binary regression," in *Proc. Annual Conf. on Inform. Sciences and Systems*, Princeton, NJ, USA, Mar. 2018.

[65] Y. Jeon, S. Hong, and N. Lee, "Supervised-learning-aided communication framework for MIMO systems with low-resolution ADCs," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7299–7313, Aug. 2018.

[66] S. Kim, M. So, N. Lee, and S. Hong, "Semi-supervised learning detector for MU-MIMO systems with one-bit ADCs," in *Proc. IEEE Int. Conf. Commun. Workshops*, Shanghai, China, May 2019.

[67] S. Kim, J. Chae, and S.-N. Hong, "Machine learning detectors for MU-MIMO systems with one-bit ADCs," *IEEE Access*, vol. 8, pp. 86 608–86 616, Apr. 2020.

[68] Y. Jeon, N. Lee, and H. V. Poor, "Robust data detection for MIMO systems with one-bit ADCs: A reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1663–1676, Mar. 2020.

[69] O. T. Demir and E. Björnson, "ADMM-based one-bit quantized signal detection for massive MIMO systems with hardware impairments," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process.*, Barcelona, Spain, May 2020, pp. 9120–9124.

[70] S. H. Song, S. Lim, G. Kwon, and H. Park, "CRC-aided soft-output detection for uplink multi-user MIMO systems with one-bit ADCs," in *Proc. IEEE Wireless Commun. and Networking Conf.*, Marrakesh, Morocco, Apr. 2019.

[71] Y. Cho and S. Hong, "One-bit Successive-cancellation Soft-output (OSS) detector for uplink MU-MIMO systems with one-bit ADCs," *IEEE Access*, vol. 7, pp. 27 172–27 182, Feb. 2019.

[72] Z. Shao, R. C. de Lamare, and L. T. N. Landau, "Iterative detection and decoding for large-scale multiple-antenna systems with 1-bit ADCs," *IEEE Wireless Commun. Letters*, vol. 7, no. 3, pp. 476–479, June 2018.

[73] S. Khobahi, N. Shlezinger, M. Soltanalian, and Y. C. Eldar, "LoRD-Net: Unfolded deep detection network with low-resolution receivers," *IEEE Trans. Signal Process.*, vol. 69, pp. 5651–5664, 2021.

[74] J. J. Bussgang, “Crosscorrelation functions of amplitude-distorted Gaussian signals,” Research Laboratory of Electronics, Massachusetts Institute of Technology, Tech. Rep. 21, 1952.

[75] O. T. Demir and E. Bjornson, “The Bussgang decomposition of nonlinear systems: Basic theory and MIMO extensions [Lecture notes],” *IEEE Signal Process. Mag.*, vol. 38, no. 1, pp. 131–136, Jan. 2021.

[76] J. Max, “Quantizing for minimum distortion,” *IRE Trans. Inf. Theory*, vol. 6, no. 1, pp. 7–12, Mar. 1960.

[77] L. V. Nguyen, D. T. Ngo, N. H. Tran, and D. H. N. Nguyen, “Learning methods for MIMO blind detection with low-resolution ADCs,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, May 2018.

[78] L. V. Nguyen, D. T. Ngo, N. H. Tran, A. L. Swindlehurst, and D. H. N. Nguyen, “Supervised and semi-supervised learning for MIMO blind detection with low-resolution ADCs,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2427–2442, Apr. 2020.

[79] Y. S. Jeon, S. N. Hong, and N. Lee, “Blind detection for MIMO systems with low-resolution ADCs using supervised learning,” in *Proc. IEEE Int. Conf. Commun.*, Paris, France, May 2017.

[80] H. Liang, W. Chung, and S. Kuo, “Coding-Aided K-means clustering blind transceiver for space shift keying MIMO systems,” *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 103–115, Jan. 2016.

[81] Y. Jeon, M. So, and N. Lee, “Reinforcement-learning-aided ML detector for uplink massive MIMO systems with low-precision ADCs,” in *Proc. IEEE Wireless Commun. and Networking Conf.*, Barcelona, Spain, Apr. 2018.

[82] Y. Jeon, H. Lee, and N. Lee, “Robust MLSD for wideband SIMO systems with one-bit ADCs: Reinforcement-Learning Approach,” in *Proc. IEEE Int. Conf. Commun. Workshops*, Kansas City, MO, USA, May 2018.

[83] S. Schibisch, S. Cammerer, S. Dorner, J. Hoydis, and S. ten Brink, “Online label recovery for deep learning-based communication through error correcting codes,” in *Proc. Int. Symp. Wireless Commun. Systems*, Lisbon, Portugal, Aug. 2018.

[84] C. M. Bishop, *Pattern Recognition and Machine Learning.* New York: Springer, 2006.

[85] *Multiplexing and Channel Coding*, 3GPP Std. TS36.212, 2012.

[86] L. V. Nguyen, D. H. N. Nguyen, and A. L. Swindlehurst, “SVM-based channel estimation and data detection for massive MIMO systems with one-bit ADCs,” in *Proc. IEEE Int. Conf.Commun. (ICC)*, Dublin, Ireland, June 2020.

[87] L. V. Nguyen, A. L. Swindlehurst, and D. H. N. Nguyen, “SVM-based channel estimation and data detection for one-bit massive MIMO systems,” *IEEE Trans. Signal Process.*, vol. 69, pp. 2086–2099, 2021.

[88] T. Joachims, "Training linear SVMs in linear time," in *Proc. the ACM SIGKDD international conference on Knowledge discovery and Data mining.* Philadelphia, PA, USA: ACM, Aug. 2006, pp. 217–226.

[89] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," Microsoft Research, Tech. Rep. MSR-TR-98-14, 1999.

[90] T. Joachims, "Making large-scale SVM learning practical," in *Advances in Kernel Methods - Support Vector Learning*, B. Scholkopf and A. Smola, Eds. MIT Press, 1998, pp. 44–56.

[91] C. W. Hsu and C. J. Lin, "A simple decomposition method for support vector machines," *Machine Learning*, vol. 46, pp. 291–314, 2002.

[92] L. Bottou and C.-J. Lin, "Support vector machine solvers," *Large scale kernel machines*, vol. 3, no. 1, pp. 301–320, 2007.

[93] M. J. F. . Garcia, J. L. Rojo-Alvarez, F. Alonso-Atienza, and M. Martinez-Ramon, "Support vector machines for robust channel estimation in OFDM," *IEEE Signal Process. Letters*, vol. 13, no. 7, pp. 397–400, July 2006.

[94] O. M. Abdul-Latif and J. Dubois, "LS-SVM detector for RMSGC diversity in SIMO channels," in *Proc. IEEE Int. Symp. on Signal Process. and Its Applications*, Sharjah, United Arab Emirates, Feb. 2007.

[95] A. Y. Chervonenkis, "Early history of support vector machines," in *Empirical Inference.* Springer, 2013, pp. 13–20.

[96] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[97] P. C. Mahalanobis, "On the generalized distance in statistics," in *Proc. National Institute of Science of India*, 1936.

[98] S. S. Keerthi and D. DeCoste, "A modified finite Newton method for fast solution of large scale linear SVMs," *Journal of Machine Learning Research*, vol. 6, pp. 341–361, Mar. 2005.

[99] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, Oct. 2011.

[100] L. V. Nguyen, D. H. N. Nguyen, and A. L. Swindlehurst, "DNN-based detectors for massive MIMO systems with low-resolution ADCs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Montreal, QC, Canada, June 2021.

[101] L. V. Nguyen, A. L. Swindlehurst, and D. H. N. Nguyen, "Linear and deep neural network-based receivers for massive MIMO systems with one-bit ADCs," *IEEE Trans. Wireless Commun.*, vol. 20, no. 11, pp. 7333–7345, Nov. 2021.

[102] L. V. Nguyen, D. H. N. Nguyen, and A. L. Swindlehurst, “Deep learning for estimation and pilot signal design in few-bit massive MIMO systems,” *submitted for journal publication (under revision), preprint arXiv:2107.11958*, 2022.

[103] J. W. Pratt, “Concavity of the log likelihood,” *J. the American Statistical Association*, vol. 76, no. 373, pp. 103–106, 1981.

[104] S. R. Bowling, M. T. Khasawneh, S. Kaewkuekool, and B. R. Cho, “A logistic approximation to the cumulative normal distribution,” *J. Industrial Engineering and Management*, vol. 2, no. 1, pp. 114–127, Mar. 2009.

[105] J. R. Hershey, J. L. Roux, and F. Weninger, “Deep unfolding: Model-based inspiration of novel deep architectures,” *arXiv:1409.2574*, 2014.

[106] N. T. Nguyen and K. Lee, “Deep learning-aided Tabu search detection for large MIMO systems,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 6, pp. 4262–4275, June 2020.

[107] M. Khani, M. Alizadeh, J. Hoydis, and P. Fleming, “Adaptive neural signal detection for massive MIMO,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5635–5648, Aug. 2020.

[108] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, Software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[109] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[110] M. Abdelghany, A. A. Farid, M. E. Rasekh, U. Madhow, and M. J. W. Rodwell, “A design framework for all-digital mmWave massive MIMO with per-antenna nonlinearities,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 9, pp. 5689–5701, Sept. 2021.

[111] Ö. Özdogan, E. Björnson, and E. G. Larsson, “Massive MIMO with spatially correlated Rician fading channels,” *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3234–3250, May 2019.

[112] N. Samuel, T. Diskin, and A. Wiesel, “Learning to detect,” *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2554–2564, May 2019.

[113] N. T. Nguyen, K. Lee, and H. Dai, “Application of deep learning to sphere decoding for large MIMO systems,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 10, pp. 6787–6803, Oct. 2021.

[114] L. V. Nguyen, N. T. Nguyen, N. H. Tran, M. Juntti, A. L. Swindlehurst, and D. H. Nguyen, "Leveraging deep neural networks for massive MIMO data detection," *IEEE Wireless Commun., preprint arXiv:2204.05350*, 2022.

[115] L. V. Nguyen, A. L. Swindlehurst, and D. H. N. Nguyen, "A variational Bayesian perspective on MIMO detection with low-resolution ADCs," *submitted to Asilomar Conference on Signals, Systems, and Computers*, 2022.

[116] V. Venkateswaran and A.-J. van der Veen, "Multichannel $\Sigma\Delta$ ADCs with integrated feedback beamformers to cancel interfering communication signals," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2211–2222, May 2011.

[117] R. M. Corey and A. C. Singer, "Spatial sigma-delta signal acquisition for wideband beamforming arrays," in *Proc. International ITG Workshop on Smart Antennas (WSA)*, Munich, Germany, Mar. 2016.

[118] D. Barac and E. Lindqvist, "Spatial sigma-delta modulation in a massive MIMO cellular system," Master's thesis, 2016.

[119] A. Nikoofard, J. Liang, M. Twieg, S. Handagala, A. Madanayake, L. Belostotski, and S. Mandal, "Low-complexity N-port adcs using 2-d $\Sigma\Delta$ noise-shaping for N-element array receivers," in *Proc. International Midwest Symposium on Circuits and Systems (MWSCAS)*, Boston, MA, USA, Aug. 2017, pp. 301–304.

[120] J. D. Krieger, C.-P. Yeang, and G. W. Wornell, "Dense delta-sigma phased arrays," *IEEE Trans. Antennas and Propagation*, vol. 61, no. 4, pp. 1825–1837, Apr. 2013.

[121] S. Rao, G. Seco-Granados, H. Pirzadeh, J. A. Nossek, and A. L. Swindlehurst, "Massive MIMO channel estimation with low-resolution spatial sigma-delta ADCs," *IEEE Access*, vol. 9, pp. 109 320–109 334, July 2021.

[122] H. Pirzadeh, G. Seco-Granados, S. Rao, and A. L. Swindlehurst, "Spectral efficiency of one-bit sigma-delta massive MIMO," *IEEE J. Select. Areas in Commun.*, vol. 38, no. 9, pp. 2215–2226, Sept. 2020.

[123] O. Castaneda, S. Jacobsson, G. Durisi, T. Goldstein, and C. Studer, "Finite-alphabet MMSE equalization for all-digital massive MU-MIMO mmWave communication," *IEEE J. Select. Areas in Commun.*, vol. 38, no. 9, pp. 2128–2141, Sept. 2020.

[124] J.-C. Chen, "One-bit MMSE equalization for all-digital massive MU-MIMO communication systems," *IEEE Systems Journal (Early Access)*, 2021.

Appendix A

Proof of Proposition 3.2

We first express $P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}}$ as follows:

$$\begin{aligned} P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}} &= \mathbb{P}\Big[\|\mathbf{y} - \check{\mathbf{y}}_k\|_2^2 \geq \|\mathbf{y} - \check{\mathbf{y}}_{k'}\|_2^2 \;\Big|\; \mathbf{x} = \check{\mathbf{x}}_k\Big] \\ &= \mathbb{P}\Big[\|\boldsymbol{\upsilon}\|_2^2 + 2\Re\{\boldsymbol{\upsilon}^H \mathbf{w}\} \leq 0\Big] \\ &= \mathbb{P}\Big[\sum_{i=1}^{N_{\mathrm{rx}}} \big(|\upsilon_i|^2 + 2\Re\{\upsilon_i^* w_i\}\big) \leq 0\Big]. \end{aligned} \tag{A.1}$$

By letting $\varepsilon_i = |\upsilon_i|^2 + 2\Re\{\upsilon_i^* w_i\}$, (A.1) becomes

$$P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}} = \mathbb{P}\Big[\sum_{i=1}^{N_{\mathrm{rx}}} \varepsilon_i \leq 0\Big]. \tag{A.2}$$

In order to approximate the probability in (A.2), we need to compute the mean and variance of ε_i. The mean of ε_i is

$$\mathbb{E}[\varepsilon_i] = \mathbb{E}\big[|\upsilon_i|^2 + 2\Re\{\upsilon_i^* w_i\}\big] = \mathbb{E}\big[|\upsilon_i|^2\big] = \sigma_{kk'}^2. \tag{A.3}$$

The variance of ε_i is given as

$$\sigma_{\varepsilon_i}^2 = \operatorname{Var}\big[|\upsilon_i|^2\big] + \operatorname{Var}\big[2\Re\{\upsilon_i^* w_i\}\big] + 2\operatorname{Cov}\big(|\upsilon_i|^2, 2\Re\{\upsilon_i^* w_i\}\big). \tag{A.4}$$

The first term in the right-hand side of (A.4) is

$$\operatorname{Var}\big[|\upsilon_i|^2\big] = \mathbb{E}\big[|\upsilon_i|^4\big] - \mathbb{E}\big[|\upsilon_i|^2\big]^2 = \sigma_{kk'}^4. \tag{A.5}$$

The second term in the right-hand side of (A.4) is

$$\operatorname{Var}\big[2\Re\{\upsilon_i^* w_i\}\big] = \operatorname{Var}\big[\upsilon_i^* w_i\big] + \operatorname{Var}\big[\upsilon_i w_i^*\big] + 2\operatorname{Cov}\big(\upsilon_i^* w_i, \upsilon_i w_i^*\big). \tag{A.6}$$

Since $\operatorname{Var}\big[\upsilon_i^* w_i\big] = \operatorname{Var}\big[\upsilon_i w_i^*\big] = \mathbb{E}\big[|\upsilon_i|^2\big] = \sigma_{kk'}^2$, and $\operatorname{Cov}\big(\upsilon_i^* w_i, \upsilon_i w_i^*\big) = 0$, we have

$$\operatorname{Var}\big[2\Re\{\upsilon_i^* w_i\}\big] = 2\sigma_{kk'}^2. \tag{A.7}$$

The last term in the right-hand side of (A.4) is

$$\operatorname{Cov}\big(|\upsilon_i|^2, 2\Re\{\upsilon_i^* w_i\}\big) = \mathbb{E}\big[|\upsilon_i|^2 2\Re\{\upsilon_i^* w_i\}\big] + \mathbb{E}\big[|\upsilon_i|^2\big]\mathbb{E}\big[2\Re\{\upsilon_i^* w_i\}\big] = 0, \tag{A.8}$$

since $\mathbb{E}\big[|\upsilon_i|^2 2\Re\{\upsilon_i^* w_i\}\big] = \mathbb{E}\big[|\upsilon_i|^2(\upsilon_i^* w_i + \upsilon_i w_i^*)\big] = 0$ and $\mathbb{E}\big[2\Re\{\upsilon_i^* w_i\}\big] = \mathbb{E}\big[\upsilon_i^* w_i\big] + \mathbb{E}\big[\upsilon_i w_i^*\big] = 0$.

Substituting the results in (A.5), (A.7), and (A.8) into (A.4) yields the variance of ε_i as

$$\sigma_{\varepsilon_i}^2 = \sigma_{kk'}^4 + 2\sigma_{kk'}^2. \tag{A.9}$$

The variables $\{\varepsilon_i\}_{i=1,\ldots,N_{\mathrm{rx}}}$ are i.i.d. because of the i.i.d. elements in $\mathbf{H}^{\mathbb{C}}$. Hence, by the central limit theorem, the variable $\sum_{i=1}^{N_{\mathrm{rx}}} \varepsilon_i$ in (A.2) can be approximated by a Gaussian random variable with mean $N_{\mathrm{rx}}\sigma_{kk'}^2$ and variance $N_{\mathrm{r}}(\sigma_{kk'}^4 + 2\sigma_{kk'}^2)$. Finally, the probability in (A.2) can be approximated as

$$P_{\check{\mathbf{x}}_k \to \check{\mathbf{x}}_{k'}} \approx \Phi\left(\frac{-N_{\mathrm{rx}}\sigma_{kk'}^2}{\sqrt{N_{\mathrm{rx}}(\sigma_{kk'}^4 + 2\sigma_{kk'}^2)}}\right) = 1 - \Phi\left(\sqrt{N_{\mathrm{rx}}/(1 + 2/\sigma_{kk'}^2)}\right). \tag{A.10}$$

Appendix B

Proof of Theorem 3.1

For two labels $\check{\mathbf{x}}_k^{\mathbb{R}}$ and $\check{\mathbf{x}}_{k'}^{\mathbb{R}}$, we can always find two disjoint index sets $\mathcal{I}_{\mathrm{c}}$ and $\mathcal{I}_{\mathrm{d}}$ such that $\check{x}_{k,i}^{\mathbb{R}} = \check{x}_{k',i}^{\mathbb{R}} \neq 0$, $\forall i \in \mathcal{I}_{\mathrm{c}}$, and $\check{x}_{k,i}^{\mathbb{R}} = -\check{x}_{k',i}^{\mathbb{R}}$ $\forall i \in \mathcal{I}_{\mathrm{d}}$. We denote $d = |\mathcal{I}_{\mathrm{d}}|$ as the Hamming distance between the two labels $\check{\mathbf{x}}_1^{\mathbb{R}}$ and $\check{\mathbf{x}}_k^{\mathbb{R}}$. Note that $d \leq N_{\mathrm{tx}}$ and $|\mathcal{I}_{\mathrm{c}}| = N_{\mathrm{tx}} - d$ for BPSK signaling. The two vectors $\mathbf{g}_1^{\mathbb{R}}$ and $\mathbf{g}_k^{\mathbb{R}}$ can now be expressed as $\mathbf{g}_k^{\mathbb{R}} = \mathbf{g}_{\mathrm{c}} + \mathbf{g}_{\mathrm{d}}$ and $\mathbf{g}_{k'}^{\mathbb{R}} = \mathbf{g}_{\mathrm{c}} - \mathbf{g}_{\mathrm{d}}$, where $\mathbf{g}_{\mathrm{c}}$ and $\mathbf{g}_{\mathrm{d}}$ are the summations of the $N_{\mathrm{tx}} - d$ and d columns of $\mathbf{H}^{\mathbb{C}}$ corresponding to the indices given in $\mathcal{I}_{\mathrm{c}}$ and $\mathcal{I}_{\mathrm{d}}$, respectively. For Rayleigh fading with unit variance, $\mathbf{g}_{\mathrm{c}}$ is $\mathcal{N}(\mathbf{0}, \frac{N_{\mathrm{tx}}-d}{2}\mathbf{I}_{2N_{\mathrm{rx}}})$ and $\mathbf{g}_{\mathrm{d}}$ is $\mathcal{N}(\mathbf{0}, \frac{d}{2}\mathbf{I}_{2N_{\mathrm{rx}}})$. The probability that $\mathrm{sign}(g_{1,i}^{\mathbb{R}}) = \mathrm{sign}(g_{k,i}^{\mathbb{R}})$ is given as

$$\mathbb{P}\big[\,\mathrm{sign}(g_{k,i}^{\mathbb{R}}) = \mathrm{sign}(g_{k',i}^{\mathbb{R}})\big] = \frac{2}{\pi}\arctan\sqrt{\frac{N_{\mathrm{tx}} - d}{d}}. \tag{B.1}$$

This is obtained by applying a result in [31], which states that if $a \sim \mathcal{N}(0, \sigma_a^2)$ and $b \sim \mathcal{N}(0, \sigma_b^2)$ then

$$\mathbb{P}\big[\,\mathrm{sign}(a + b) = \mathrm{sign}(a - b)\big] = \frac{2}{\pi}\arctan\frac{\sigma_a}{\sigma_b}. \tag{B.2}$$

Due to the independence between the events $\mathrm{sign}(g_{k,i}^{\mathbb{R}}) = \mathrm{sign}(g_{k',i}^{\mathbb{R}})$, for $i = 1, 2, \ldots, 2N_{\mathrm{rx}}$, the result in (3.35) thus follows.

Appendix C

Proof of Proposition 3.4

Without loss of generality, we assume that $\check{\mathbf{x}}_1^{\mathbb{R}} = [\mathbf{1}_{N_{\mathrm{tx}}}^T, \mathbf{0}_{N_{\mathrm{tx}}}^T]^T$ was transmitted. Denote E_k, $1 < k \leq K$, as the event $\check{\mathbf{y}}_1 = \check{\mathbf{y}}_k$. The detection error event E is then defined as $E = \bigcup_{k>1} E_k$. We want to find the VER given event E and subsequently prove that $P_{\rho\to\infty}^{\mathrm{ver}} \leq \frac{1}{2}\sum_{k>1}^{K} \mathbb{P}(E_k)$. We note that $E_2, \ldots, E_K$ are not necessarily mutually exclusive nor independent. However, we can combine $E_2, \ldots, E_K$ into larger events $G_1, \ldots, G_L$ that are mutually exclusive. Herein, the rule for forming G_ℓ is as follows:

1. If E_k is mutually exclusive with all other events, then $E_k \subset G_1$.
2. If a pair of events E_k and E_m intersect, i.e., $E_k \cap E_m \neq \varnothing$, but $E_k \cup E_m$ is mutually exclusive with all other events, then $(E_k \cup E_m) \subset G_2$.
3. $G_3, \ldots, G_L$ are then formed in a similar fashion.

Certainly, if $E_k \subset G_\ell$, then $E_k \cap G_{\ell'} = \varnothing$, for $\ell' \neq \ell$. This combining strategy effectively partitions E into mutually exclusive events $G_1, \ldots, G_L$. The VER is calculated as:

1. If event $E_k \subset G_1$ has occurred, the receiver would erroneously pick the detected vector

$\hat{\mathbf{x}}_k^{\mathbb{R}} \neq \check{\mathbf{x}}_1^{\mathbb{R}}$ with a probability of 1/2, i.e., VER $= 1/2$.

2. For any two events $E_k, E_m \subset G_2$ and $E_k \cap E_m \neq \varnothing$, we consider the following three partitions of $E_k \cup E_m$:

   - If $E_k \cap E_m^{\mathrm{c}}$ has occurred, VER $= 1/2$.
   - If $E_k^{\mathrm{c}} \cap E_m$ has occurred, VER $= 1/2$.
   - If $E_k \cap E_m$ has occurred, the receiver would erroneously pick the detected vector as either $\hat{\mathbf{x}}_k^{\mathbb{R}}$ or $\hat{\mathbf{x}}_m^{\mathbb{R}}$ with a probability of 2/3, i.e., VER $= 2/3$.

   We then have

$$\begin{aligned} & \frac{1}{2}\mathbb{P}[E_k \cap E_m^{\mathrm{c}}] + \frac{1}{2}\mathbb{P}[E_k^{\mathrm{c}} \cap E_m] + \frac{2}{3}\mathbb{P}[E_k \cap E_m] \\ \leq \; & \frac{1}{2}\mathbb{P}[E_k \cap E_m^{\mathrm{c}}] + \frac{1}{2}\mathbb{P}[E_k^{\mathrm{c}} \cap E_m] + \mathbb{P}[E_k \cap E_m] = \frac{1}{2}\mathbb{P}[E_k] + \frac{1}{2}\mathbb{P}[E_m]. \end{aligned} \tag{C.1}$$

3. The same principle of partitioning can be applied for events in $G_3, \ldots, G_L$ to calculate the VER.

Therefore, $P_{\rho\to\infty}^{\mathrm{ver}}$ is upper-bounded as

$$\begin{aligned} P_{\rho\to\infty}^{\mathrm{ver}} & \leq \sum_{E_k \subset G_1} \frac{1}{2}\mathbb{P}[E_k] + \sum_{E_k \subset G_2} \frac{1}{2}\mathbb{P}[E_k] + \ldots \\ & = \frac{1}{2}\sum_{k>1}^{K} \mathbb{P}[E_k]. \end{aligned} \tag{C.2}$$

The inequality presented in the proposition follows by combining the result in Theorem 3.1 and noting that there are $\binom{N_{\mathrm{tx}}}{d}$ labels with Hamming distance d from $\check{\mathbf{x}}_1^{\mathbb{R}}$. If the error event E is comprised of only mutual events $E_2, \ldots, E_K$, the inequality (C.2) becomes $P_{\rho\to\infty}^{\mathrm{ver}} = \sum_{k=2}^{K} \frac{1}{2}\mathbb{P}[E_k]$. Thus, the VER upper-bound becomes tight in this case.

Appendix D

Explanation for the susceptibility of ML detection at high SNRs with imperfect CSI

The ML detection method of [31] is defined as

$$\hat{\mathbf{x}}_{\mathrm{d},m}^{\mathtt{ML}} = \underset{\mathbf{x}^{\mathbb{C}} \in (\mathcal{M}^{\mathbb{C}})^U}{\arg\max} \underbrace{\prod_{i=1}^{2N} \Phi\left(\sqrt{2\varrho} y_{\mathrm{d},m,i} \hat{\mathbf{h}}_{\mathrm{d},i}^T \mathbf{x}\right)}_{\mathcal{P}(\mathbf{x})}, \tag{D.1}$$

where $\mathbf{x} = [\Re\{\mathbf{x}^{\mathbb{C}}\}^T, \Im\{\mathbf{x}^{\mathbb{C}}\}^T]^T$ and $\mathcal{P}(\mathbf{x})$ is the likelihood function. It is clear that as $\varrho \to \infty$, we have

$$\begin{cases} \Phi\left(\sqrt{2\varrho} y_{\mathrm{d},m,i} \hat{\mathbf{h}}_{\mathrm{d},i}^T \mathbf{x}\right) \to 0 \text{ if } y_{\mathrm{d},m,i} \hat{\mathbf{h}}_{\mathrm{d},i}^T \mathbf{x} < 0, \\ \Phi\left(\sqrt{2\varrho} y_{\mathrm{d},m,i} \hat{\mathbf{h}}_{\mathrm{d},i}^T \mathbf{x}\right) \to 1 \text{ if } y_{\mathrm{d},m,i} \hat{\mathbf{h}}_{\mathrm{d},i}^T \mathbf{x} > 0. \end{cases}$$

This means, as $\varrho \to \infty$, $\mathcal{P}(\mathbf{x}) = 0$ if there exists at least one index i such that $y_{\mathrm{d},m,i} \hat{\mathbf{h}}_{\mathrm{d},i}^T \mathbf{x} < 0$ and $\mathcal{P}(\mathbf{x}) = 1$ if $y_{\mathrm{d},m,i} \hat{\mathbf{h}}_{\mathrm{d},i}^T \mathbf{x} > 0$ for all i.

Now, suppose that a vector $\mathbf{x}^{\star\mathbb{C}}$ was transmitted and let $\mathbf{x}^{\star} = [\Re\{\mathbf{x}^{\star\mathbb{C}}\}^T, \Im\{\mathbf{x}^{\star\mathbb{C}}\}^T]^T$. If the CSI is perfectly known, i.e., $\hat{\mathbf{h}}_{\mathrm{d},i} = \mathbf{h}_{\mathrm{d},i}$, we have $y_{\mathrm{d},m,i}\hat{\mathbf{h}}_{\mathrm{d},i}^T\mathbf{x}^{\star} > 0$ for all i because $y_{\mathrm{d},m,i} = \mathrm{sign}(\mathbf{h}_{\mathrm{d},i}^T\mathbf{x}^{\star}) = \mathrm{sign}(\hat{\mathbf{h}}_{\mathrm{d},i}^T\mathbf{x}^{\star})$ as $\varrho \to \infty$. In other words, $\mathcal{P}(\mathbf{x}^{\star}) = 1$ if the CSI is perfectly known at infinite SNR. However, if the CSI is not known perfectly, i.e., $\hat{\mathbf{h}}_{\mathrm{d},i} \neq \mathbf{h}_{\mathrm{d},i}$, there is a non-zero probability that $y_{\mathrm{d},m,i} = \mathrm{sign}(\mathbf{h}_{\mathrm{d},i}^T\mathbf{x}^{\star}) \neq \mathrm{sign}(\hat{\mathbf{h}}_{\mathrm{d},i}^T\mathbf{x}^{\star})$, which means $y_{\mathrm{d},m,i}\,\mathrm{sign}(\hat{\mathbf{h}}_{\mathrm{d},i}^T\mathbf{x}^{\star}) < 0$. This causes $\mathcal{P}(\mathbf{x}^{\star}) = 0$. For any $\mathbf{x} \neq \mathbf{x}^{\star}$, it is possible that $y_{\mathrm{d},m,i} = \mathrm{sign}(\mathbf{h}_{\mathrm{d},i}^T\mathbf{x}^{\star}) \neq \mathrm{sign}(\hat{\mathbf{h}}_{\mathrm{d},i}^T\mathbf{x})$, which also leads to $\mathcal{P}(\mathbf{x}) = 0$. Hence, detection errors occur. The above explanation is argued at infinite SNR, but it is also valid for high SNRs because $\Phi(t)$ approaches 0 very fast.

To remove the product in (D.1), one may argue to transform the function $\mathcal{L}(\mathbf{x})$ into a sum of log functions as follows:

$$\hat{\mathbf{x}}_{\mathrm{d},m}^{\mathtt{ML}} = \underset{\mathbf{x}^{\mathbb{C}} \in (\mathcal{M}^{\mathbb{C}})^U}{\arg\max} \underbrace{\sum_{i=1}^{2N} \log \Phi\left(\sqrt{2\varrho}\, y_{\mathrm{d},m,i}\hat{\mathbf{h}}_{\mathrm{d},i}^T\mathbf{x}\right)}_{\mathcal{P}(\mathbf{x})}. \tag{D.2}$$

However, the function $\mathcal{P}(\mathbf{x})$ in (D.2) still depends on $\Phi(\cdot)$ and can involve $\log(0)$. The proposed SVM-based data detection method is robust against imperfect CSI since it does not depend on the $\Phi(\cdot)$ function and information about the SNR is not required either.

We note that the OSD method in [61] is also robust against imperfect CSI thanks to the use of the approximation $1 - \Phi(t) \approx \frac{1}{2}e^{-0.374t^2 - 0.777t}$ for non-negative t. This approximation helps remove the effect of $\log \Phi(\cdot)$ in (D.2) since $\log e^a = a$. However, the OSD method has higher computational complexity than the proposed SVM-based methods.

Appendix E

Proof of Proposition 5.1

Since $\mathbf{x}_m$ is the m^{th} nearest symbol vector, we have the following condition:

$$\|\mathbf{x}_1 - \tilde{\mathbf{x}}\|^2 < \ldots < \|\mathbf{x}_{m-1} - \tilde{\mathbf{x}}\|^2 < \|\mathbf{x}_m - \tilde{\mathbf{x}}\|^2 < \|\mathbf{x} - \tilde{\mathbf{x}}\|^2 \tag{E.1}$$

for any $\mathbf{x} \notin \mathcal{X}_m$.

We prove the proposition by contradiction. Suppose that $\mathbf{x}_m$ is not a neighbor of $\mathcal{X}_{m-1}$, i.e., $\mathbf{x}_m \notin \mathcal{N}(\mathcal{X}_{m-1})$ or $d_{\min}(\mathbf{x}_m, \mathcal{X}_{m-1}) > 1$. For the sake of simplicity, we consider the case where $d_{\min}(\mathbf{x}_m, \mathcal{X}_{m-1}) = 2$. Proof for the other cases where $d_{\min}(\mathbf{x}_m, \mathcal{X}_{m-1}) > 2$ can be accomplished similarly.

Let $\mathbf{x}_p \in \mathcal{X}_{m-1}$ with $p \in \{1, 2, \ldots, m-1\}$ be a symbol vector such that $d(\mathbf{x}_p, \mathbf{x}_m) = 2$. Without loss of generality, we can always assume that the two position indices at which the differences occur are 1 and 2, i.e.,

$$\begin{cases} x_{m,1} \neq x_{p,1} \\ x_{m,2} \neq x_{p,2} \\ x_{m,i} = x_{p,i} \ \forall i \in \{3, \dots, 2K\}. \end{cases} \tag{E.2}$$

Now, we consider two other symbol vectors $\mathbf{x}' = [x'_1, \dots, x'_{2K}]^T$ and $\mathbf{x}'' = [x''_1, \dots, x''_{2K}]^T$ such that

$$\begin{cases} x'_1 = x_{m,1} \neq x_{p,1} = x''_1 \\ x'_2 = x_{p,2} \neq x_{m,2} = x''_2 \\ x'_i = x''_i = x_{p,i} = x_{m,i} \ \forall i \in \{3, \dots, 2K\}. \end{cases} \tag{E.3}$$

Hence, $\mathbf{x}'$ and $\mathbf{x}''$ are the two symbol vectors satisfying $d(\mathbf{x}', \mathbf{x}_m) = d(\mathbf{x}'', \mathbf{x}_m) = 1$. In other words, both $\mathbf{x}'$ and $\mathbf{x}''$ are neighbors of $\mathbf{x}_m$.

If $\mathbf{x}' \in \mathcal{X}_{m-1}$ and/or $\mathbf{x}'' \in \mathcal{X}_{m-1}$, then $d_{\min}(\mathbf{x}_m, \mathcal{X}_{m-1}) = 1$ because $\mathbf{x}_m$ is a neighbor of both $\mathbf{x}'$ and $\mathbf{x}''$, which is contradicted by the assumption that $d_{\min}(\mathbf{x}_m, \mathcal{X}_{m-1}) = 2$. Thus, $\mathbf{x}_m$ is a neighbor of $\mathcal{X}_{m-1}$, i.e, $\mathbf{x}_m \in \mathcal{N}(\mathcal{X}_{m-1})$.

If $\mathbf{x}' \notin \mathcal{X}_{m-1}$ and $\mathbf{x}'' \notin \mathcal{X}_{m-1}$, we have

$$|x_{m,1} - \tilde{x}_1|^2 = |x'_1 - \tilde{x}_1|^2 > |x_{p,1} - \tilde{x}_1|^2. \tag{E.4}$$

Adding both sides of (E.4) with $|x_{m,2} - \tilde{x}_2|^2$ yields

$$|x_{m,1} - \tilde{x}_1|^2 + |x_{m,2} - \tilde{x}_2|^2 > |x_{p,1} - \tilde{x}_1|^2 + |x_{m,2} - \tilde{x}_2|^2,$$

which can be rewritten as

$$|x_{m,1} - \tilde{x}_1|^2 + |x_{m,2} - \tilde{x}_2|^2 > |x''_1 - \tilde{x}_1|^2 + |x''_2 - \tilde{x}_2|^2 \tag{E.5}$$

because $x_{p,1} = x''_1$ and $x_{m,2} = x''_2$. The inequality in (E.5) indicates that $\|\mathbf{x}_m - \tilde{\mathbf{x}}\|^2 > \|\mathbf{x}'' - \tilde{\mathbf{x}}\|^2$, which means $\mathbf{x}''$ is closer to $\tilde{\mathbf{x}}$ than $\mathbf{x}_m$, or in other words, $\mathbf{x}_m$ is not the m^{th} nearest symbol vector of $\tilde{\mathbf{x}}$. This is contradicted by (E.1).