



FACHBEREICH 1 - PHYSICS AND ELECTRONICAL ENGINEERING  
INSTITUTE FOR TELECOMMUNICATIONS AND HIGH-FREQUENCY  
TECHNIQUES (ITH) DEPARTMENT OF COMMUNICATIONS  
ENGINEERING P.O. BOX 33 04 40 D-28334 BREMEN DEPARTMENT  
OF PHYSICS / ELECTRICAL ENGINEERING

# **APPLICATIONS OF RESERVOIR COMPUTING IN PHY-LAYER PROCESSING**

AUTHOR: SHAWON KUMAR MONDAL ID (ID: 6001628)  
THESIS SUPERVISOR: TIM DUE

## **Declaration**

I confirm that I have read and understood the University's Academic Integrity Policy.

I confirm that I have acted honestly, ethically, and professionally in conduct leading to assessment for the programme of study.

I confirm that I have not copied material from another source, committed plagiarism, nor fabricated, falsified, or embellished data when completing the attached piece of work. I confirm that I have not copied material from another source, nor colluded with any other student in the preparation and production of this work.

Signed: Shawon Kumar Mondal

Date: August 19, 2025

# Abstract

Deep learning (DL) is gaining widespread use in various layers of wireless communication systems, especially in the physical layer. DL demonstrates strong performance in tasks like channel equalization, receiver signal processing, and symbol detection.

The main aim of this thesis is to study and evaluate the effectiveness of machine learning (ML)-based techniques, particularly focusing on a specialized form of Reservoir Computing (RC) known as Echo State Networks (ESNs). This work specifically explores Deep Echo State Networks (DESNs), which are multi layered extensions of ESNs.

Reservoir computing refers to a class of recurrent neural networks (RNNs) in which most of the weights are randomly fixed, and only a small set typically the output layer is trained. This setup allows for fast training and efficient handling of complex temporal patterns.

Compared to other ML models like Convolutional Neural Networks (CNNs) and Graph Neural Networks (GNNs), DESNs have shown superior capabilities in challenging channel conditions, such as those modeled by Time Delay Channel(TDC) or Clustered Delay Line (CDL) framework.

In wireless communication, signals often suffer from distortion as they travel through the air. This issue becomes more severe when nonlinearities are introduced by components like the transmitter's power amplifier. As a result, accurate symbol detection becomes a critical task. Traditional systems rely heavily on precise channel estimation, especially in MIMO-OFDM frameworks.

This thesis introduces a research based approach that eliminates the need for explicit channel estimation by using ESNs. The ESN acts as a black-box model that can learn and predict nonlinear dynamics in wireless systems. Simulation results confirm that this method outper-

forms traditional detection strategies, particularly under severe distortion conditions.

# Contents

<b>Abstract</b>	<b>ii</b>
<b>Contents</b>	<b>iv</b>
<b>List of Figures</b>	<b>vii</b>
<b>Acronyms</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Purpose . . . . .	2
1.2 Motivation and Real-World Problem . . . . .	3
1.3 Research Questions and Objectives . . . . .	5
<b>2 Literature Review</b>	<b>7</b>
2.1 Foundations of Modern Wireless Communication . . . . .	8
2.1.1 Orthogonal Frequency Division Multiplexing (Orthogonal Frequency Division Multiplexing (OFDM))	9
2.1.2 Introduction to Single-Input Single-Output (SISO), Single-Input Multiple-Output (SIMO), and Multiple-Input Multiple-Output (MIMO) Systems . . . . .	10
2.1.3 MIMO-OFDM Systems . . . . .	12
2.2 Reservoir Computing and Echo State Network (ESN) Applications Physical Layer Processing . . . . .	13
2.2.1 Channel Estimation using ESNs . . . . .	14
2.2.2 Symbol Detection . . . . .	14
2.2.3 Other Applications . . . . .	15
2.3 Physical Layer Processing through ML/DL Models . . . . .	15
2.3.1 Constellation Identification and Modulation Classification . . . . .	16
2.3.2 Channel Estimation and Equalization . . . . .	16

2.3.3	Symbol Detection . . . . .	17
2.3.4	Spectrum Allocation and Interference Management	17
2.3.5	Nonlinear Channel Equalization in LEO Satellites	18
2.3.6	Adaptive Modulation and SNR Estimation . . . . .	18
2.3.7	Real Time Adaptation in Dynamic Channels . . . . .	19
2.4	Performance Evaluations and Comparisons . . . . .	20
2.4.1	Comparison ESN with other Machine Learning (ML) models . . . . .	20
2.4.2	Hardware Efficiency . . . . .	22
2.5	Synthesis of Research Gaps . . . . .	23
2.5.1	Summary of Research Gaps . . . . .	24
2.6	2.6 Research Direction and Implementation Scope . . . . .	25
<b>3</b>	<b>Echo State Networks: Architecture, Analysis, and Application in OFDM Systems</b>	<b>27</b>
3.1	Reservoir Computing (Reservoir Computing (RC)) and Echo State Networks (ESNs) . . . . .	28
3.1.1	Deep Echo State Networks (DESN) for Symbol Detection . . . . .	34
3.2	System Architecture . . . . .	35
3.2.1	System Model 1 . . . . .	36
3.2.2	System Model 2 . . . . .	36
3.3	Benefits of ESNs for Symbol Detection . . . . .	45
3.3.1	Key Advantages . . . . .	45
3.3.2	Implementation Considerations . . . . .	46
3.3.3	Challenges and Limitations . . . . .	47
3.4	Baseline Models for Comparison . . . . .	47
3.4.1	Classical Signal Processing Methods . . . . .	48
3.4.2	Deep Learning-Based Baselines . . . . .	49
3.4.3	Motivation and Comparison Strategy . . . . .	50
<b>4</b>	<b>Results and Analysis</b>	<b>52</b>
4.1	Evaluation Strategy and First Pipeline . . . . .	53
4.1.1	Results for System Model 1 . . . . .	55
4.2	System Model II: Results . . . . .	57
4.2.1	System Model II: Stage 1 OFDM SISO under Additive White Gaussian Noise (AWGN) . . . . .	57
4.2.2	SISO Block-Fading (16 Quadrature Amplitude Modulation (QAM))Setup and Result . . . . .	61

4.2.3	SIMO Block-Fading (16-QAM) . . . . .	63
4.2.4	MIMO $2 \times 2$ Block-Fading (16-QAM) . . . . .	64
4.2.5	MIMO $4 \times 8$ (16-QAM) with Low-Density Parity- Check (LDPC): Training-Signal-to-Noise Ratio (SNR) .	66
4.2.6	MIMO $4 \times 8$ with Clustered Delay Line (CDL) (16- QAM) and LDPC: Training-SNR . . . . .	69
4.2.7	MIMO $2 \times 2$ with ML-Based Detectors: Model Com- parison . . . . .	72
4.3	ESN Hyperparameter Selection (System Model II) . . . . .	73
<b>Conclusion and Future Directions</b>		<b>76</b>
4.4	Summary of key findings . . . . .	76
4.5	Contributions to the field . . . . .	76
4.6	Study limitations . . . . .	76
4.7	Recommendations for future research . . . . .	76
4.8	Final Thoughts . . . . .	76
<b>Bibliography</b>		<b>77</b>
<b>Appendix</b>		<b>80</b>
<b>Appendix 1 - Code Embed</b>		<b>81</b>

# List of Figures

1.1	Communication scenario illustrating intended AP-node communication affected by interfering APs and nodes. Adapted from [1]. . . . .	3
2.1	Combined time/frequency domain view of OFDM signal. Adapted from [2]. . . . .	10
2.2	Illustration of SISO, SIMO, MISO, and MIMO antenna configurations. Source: Adapted from Wikimedia Commons [3]. . . . .	12
3.1	Mackey–Glass prediction with ESN. Black: target series; red: ESN free run after training (vertical dotted line marks the train/test split). . . . .	30
3.2	Echo State Network (ESN) architecture comprising input, dynamic reservoir, and trainable readout layers. . . . .	31
3.3	Deep Echo State Network (DESN) architecture comprising multiple stacked reservoirs with intermediate input/output layers. Each reservoir receives input from the previous layer, enabling hierarchical temporal modeling. Adapted from [4]. . . . .	35
3.4	General Framework of the MIMO-OFDM System Architecture. . . . .	36
3.5	<b>System Model 2.</b> ESN-based time-domain detector with minimal front-end processing: pilot+IDFT → Cyclic Prefix (CP)+Power Amplifier (PA) → MIMO channel → Re/Im to ESN → Discrete Fourier Transform (DFT) to obtain $\tilde{\mathbf{X}}_{\text{ESN}}$ . . . . .	37
4.1	BER performance for different demappers in System Model 1. . . . .	56
4.2	Constellation diagrams for different demappers in System Model 1, coloured by predicted symbol index. . . . .	56

4.3	Uncoded Bit Error Rate (BER) vs. $E_b/N_0$ for SISO 4-QAM under AWGN with soft PA: Perfect Channel State Information (CSI), Least Squares (LS), Linear Minimum-Mean-Square Error (LMMSE), ESN (matched and fixed-train@12 dB), and Extreme Learning Machine (ELM) (matched and fixed-train). . . . .	59
4.4	SISO, 16-QAM, block-fading with soft PA: uncoded BER vs. $E_b/N_0$ for Perfect-CSI ZF, Minimum-Mean-Square Error (MMSE), LS-ZF, and ESN (matched and fixed-train at 12 dB). . . . .	61
4.5	Average per-subcarrier capacity $C = \mathbb{E}_k[\log_2(1 + \text{SNR}_k)]$ from the same $\mathbf{H}$ ensemble for the SISO block-fading case. . . . .	62
4.6	SIMO $1 \times 2$ : uncoded BER vs. $E_b/N_0$ . <b>All linear baselines use ZF detection</b> , differing only in the channel estimate: <i>Perfect-CE</i> , <i>LS-CE</i> , and <i>LMMSE-CE</i> . Learning curves are shown for <i>ESN (matched)</i> and <i>ESN (train @ 12 dB)</i> . . . . .	63
4.7	SIMO $1 \times 2$ : average per-subcarrier capacity, $C = \mathbb{E}_k\left[\log_2\left(1 + \frac{\Pi}{N_0} \ \mathbf{h}_k\ _2^2\right)\right]$ , obtained from the same channel ensemble. . . . .	64
4.8	$2 \times 2$ MIMO, 16-QAM, nonlinear PA: uncoded BER vs. $E_b/N_0$ for Perfect ZF, LS-ZF, MMSE, and ESN (matched and fixed-train@12 dB). . . . .	65
4.9	$2 \times 2$ MIMO capacity (average per subcarrier) and fraction of usable rank $\geq 2$ obtained from the same $\mathbf{H}_k$ ensemble. . . . .	66
4.10	MIMO $4 \times 8$ : uncoded (pre-LDPC) BER vs. $E_b/N_0$ for Perfect ZF, LS-ZF, MMSE, and ESN (matched and fixed @12 dB). . . . .	67
4.11	MIMO $4 \times 8$ : coded (post-LDPC) BER vs. $E_b/N_0$ . LDPC decoding compresses performance gaps among linear baselines but ESN remains non-competitive. . . . .	68
4.12	MIMO $4 \times 8$ : ESN-only comparison (pre- vs. post-LDPC) for matched and fixed 12 dB training. LDPC reduces BER somewhat but cannot close the gap to baselines. . . . .	68
4.13	$4 \times 8$ MIMO capacity (average per subcarrier) and fraction of usable rank $\geq 4$ obtained from per-subcarrier SVD. . . . .	69
4.14	$4 \times 8$ MIMO over CDL-B: uncoded vs. coded (LDPC) BER with Log-Likelihood Ratio (LLR) calibration; comparison of ESN demapper and linear MMSE. . . . .	70

4.15	$4 \times 8$ MIMO over CDL–B: average per-subcarrier capacity and fraction of subcarriers with usable rank $\geq 4$ . . . . .	71
4.16	MIMO $2 \times 2$ (block fading): uncoded BER vs. $E_b/N_0$ for ESN, Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Feed-Forward Neural Network (FFNN), ELM, and classical baselines. . . . .	73
4.17	One-at-a-time ESN sweeps: reservoir size, spectral radius $\rho$ , input scaling, sparsity, and teacher scaling (BER vs. $E_b/N_0$ ). . . . .	75

*Shawon Kumar Mondal (6001628)*

# Acronyms

**5G** Fifth Generation.

**6G** Sixth Generation.

**AWGN** Additive White Gaussian Noise.

**BER** Bit Error Rate.

**CDL** Clustered Delay Line.

**CNN** Convolutional Neural Network.

**CP** Cyclic Prefix.

**CRJ** Cycle Reservoir with Jumps.

**CSI** Channel State Information.

**DESN** Deep Echo State Network.

**DFT** Discrete Fourier Transform.

**DL** Deep Learning.

**ELM** Extreme Learning Machine.

**ESN** Echo State Network.

**ESP** Echo State Property.

**FDM** Frequency Division Multiplexing.

**FFNN** Feed-Forward Neural Network.

**FFT** Fast Fourier Transform.

**GNN** Graph Neural Network.

**LDPC** Low-Density Parity-Check.

**LEO** Low Earth Orbit.

**LLR** Log-Likelihood Ratio.

**LMMSE** Linear Minimum-Mean-Square Error.

**LS** Least Squares.

**LSTM** Long Short-Term Memory.

**MIMO** Multiple-Input Multiple-Output.

**ML** Machine Learning.

**MMSE** Minimum-Mean-Square Error.

**NN** Neural Network.

**NTN** Non-Terrestrial Network.

**OFDM** Orthogonal Frequency Division Multiplexing.

**PA** Power Amplifier.

**PAPR** Peak-to-Average Power Ratio.

**PDP** Power Delay Profile.

**QAM** Quadrature Amplitude Modulation.

**RC** Reservoir Computing.

**RNN** Recurrent Neural Network.

**SCR** Simple Cycle Reservoir.

**SER** Symbol Error Rate.

**SIMO** Single-Input Multiple-Output.

**SISO** Single-Input Single-Output.

**SNR** Signal-to-Noise Ratio.

**SVD** Singular Value Decomposition.

**TN** Terrestrial Network.

**ZF** Zero Forcing.

[ type=acronym, title=List of Abbreviations, style=long3col ]

# 1 | Introduction

Exponential growth in wireless communication technologies contributed towards more complex system architectures to achieve higher throughput, lower latency, and reliability demands. At its core is the integration of Orthogonal Frequency Division Multiplexing (OFDM) with Multiple Input Multiple Output (MIMO) systems. The compact union of MIMO and OFDM, allows the wireless system to provide high spectral efficiency over broadband channels with less transceiver architected simplifications. This is one of the prime reasons that MIMO-OFDM is not only applied in the majority of contemporary wireless systems but is also forecast for upcoming networks. MIMO adds additional spatial degrees of freedom to the transmission link which allows the system with flexible transmission modes, e.g., spatial-multiplexing and transmit-receive diversity.

This integration utilizes spatial and frequency diversity to increase spectral efficiency and reliability significantly in multipath settings. Equalization is made simple through the breakup of frequency-selective wideband channel into many flat-fading subchannels by OFDM, thus decreasing inter-symbol interference [5].

While offering all these benefits, OFDM poses critical implementation issues. The main among them is its very high Peak-to-Average Power Ratio (Peak-to-Average Power Ratio (PAPR)), which requires Power Amplifiers (PAs) to be operated in their linear region. To achieve better power efficiency, PAs must be driven into saturation and so add

nonlinear distortions that modify amplitude and phase properties of forwarded signals. Such distortions undermine orthogonality among OFDM subcarriers and significantly degrade symbol detection at the receiver [6].

In addition, efficiency in MIMO-OFDM systems relies greatly on obtaining accurate Channel State Information (CSI) at the receiver side. Acquisition of CSI, however, is computationally complex under conditions involving high mobility, rapidly fading channels, and nonlinear distortion [7]. Least Squares (LS) and Linear Minimum Mean Square Error (LMMSE) estimators commonly used previously become unreliable and even expensive in such an environment. With wireless systems trending towards more high-dimensional configurations, the practicality of using these methods further wane away. In order to avoid the constraints of traditional detection and estimation frameworks, researchers have more and more resorted to Machine Learning (ML), and more recently to biologically inspired neural architectures. RC is one approach that is emerging as a viable alternative. RC is a computationally efficient framework for modeling dynamic temporal systems with fixed recurrent connections in a high-dimensional reservoir and linearly adjustable output alone. Echo State Networks (ESNs), a popular family of RC, especially perform well in handling nonlinear sequence prediction tasks due to their memory-rich internal dynamics and low training complexity [8, 9].

## **1.1 Purpose**

This thesis examines the application of Echo State Networks ESNs, to symbol detection in nonlinear SISO,SIMO and also MIMO-OFDM communications. The aim is to show the viability of a decoding framework that is carried out without direct channel estimation through learning input-output dynamics from distorted observations directly. This re-

search evaluates the proposed reservoir computing framework across SISO, SIMO, and MIMO configurations, and compares its performance with classical linear estimators with different ML models.

## 1.2 Motivation and Real-World Problem

Practical applications such as high-speed vehicular networks, drone relays, and non-terrestrial networks (NTNs) are subject to rapidly varying propagation channels and hardware-based impairments, rendering conventional pilot-based detection methods unpredictable. Drawn from instantaneous CSI notifications in such situations, it usually latency along with unnecessary computational and energy overheads, which are not desirable for energy-limited, embedded, or mobile devices under consideration for future 6G implementations.

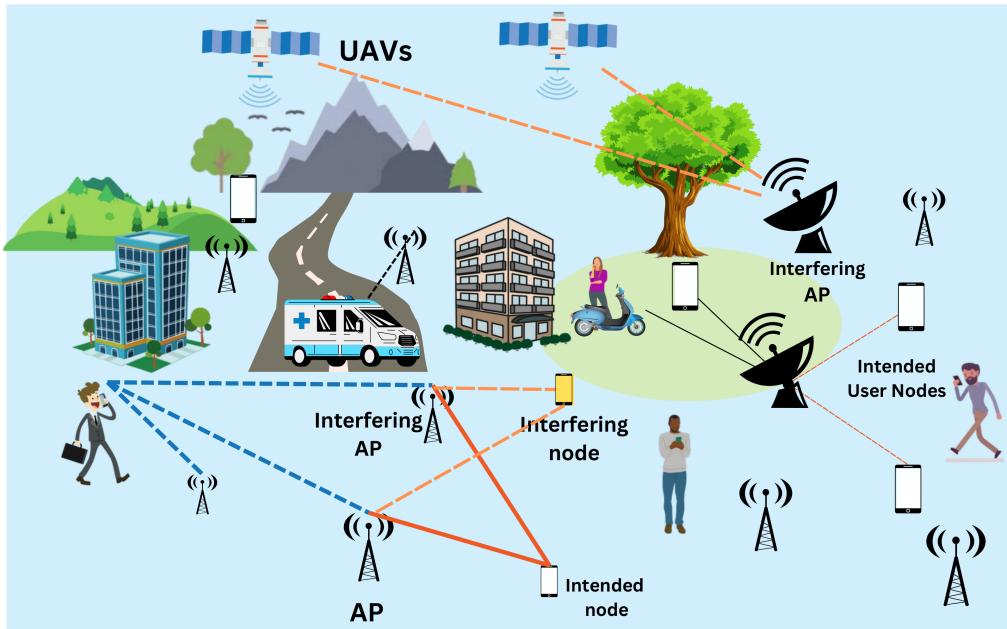


Figure 1.1: Communication scenario illustrating intended AP-node communication affected by interfering APs and nodes. Adapted from [1].

These challenges have motivated researchers to explore data driven, model agnostic methods capable of learning directly from received signals. While neural networks such as CNNs and RNNs have been proposed for such tasks, they often suffer from long training times, memory

requirements, and convergence difficulties [10, 11].

As an alternative, ESNs a form of Reservoir Computing offer several advantages: they retain the temporal memory of RNNs, but train only the output layer, making them lightweight and faster to deploy in real-time systems. Their ability to handle dynamic inputs and nonlinear effects, with significantly lower complexity, makes them an excellent candidate for physical layer processing tasks such as symbol detection [5, 12].

Moreover, OFDM signals are inherently structured in both time and frequency domains due to the sequential nature of their IFFT/FFT modulation processes. This structure aligns naturally with the working principles of ESNs, which are especially effective in time-series prediction and pattern recognition. Recent studies have shown that integrating the time frequency structure of OFDM into reservoir computing frameworks such as the deep time-frequency Echo State Network can lead to significant improvements in symbol detection, even under harsh channel conditions and nonlinear system impairments [13].

ESNs have also been extensively used in various time-series domains, validating their relevance to wireless systems. Originally introduced by Jaeger, they have been applied successfully in tasks like time series forecasting, [14] speech recognition, [15] noise modeling, [14] dynamic pattern classification, [16] reinforcement learning, [17] and language modeling [18]. These tasks all share sequential, temporally correlated data structures, similar to OFDM symbols. Furthermore, the ESN architecture has evolved to include enhancements such as leaky integrator neurons, hierarchical and small-world topologies, and hybrid readout mechanisms, making it increasingly adaptable to communication problems.

By combining these insights, this thesis investigates the application of ESNs for robust symbol detection in nonlinear MIMO OFDM sys-

tems. The goal is to evaluate their potential for achieving low complexity, low latency, and high performance detection under practical conditions relevant to future wireless systems.

### **1.3 Research Questions and Objectives**

In order to solve the above mentioned challenge, this research explores and investigates a decoding method based on ESNs. The main objectives of this work are:

- To provide an emblem recognition scheme that repels direct CSI estimation by ESNs in nonlinear and distorted MIMO-OFDM systems.
- To optimize and design ESN architectures by adjusting important parameters like reservoir size, spectral radius, and inter-layer connectivity to realize better nonlinear detection capability.
- To contrast the performance of ESNs with baseline detection methods (e.g., LS, LMMSE) and shallow neural networks in terms of BER, and computational complexity.
- To predict the scalability and hardware implementation viability of the new method for real-time and embedded wireless systems.

Based on these objectives, the following research questions are formulated:

1. Is accurate symbol detection possible in time-variant, nonlinear SISO,SIMO,MIMO-OFDM channels using a ESN-based receiver with or without direct CSI?
2. Does ESNs demonstrate greater performance than shallow dense-layer neural networks and conventional estimators under different SNR and distortion environments?
3. How do architectural parameters like spectral radius, dropout rate,

and reservoir depth control the performance of Deep Echo State Networks (DESNs)?

4. Is the suggested decoding scheme feasible for real-world deployment in terms of energy efficiency, inference latency, and hardware integration?

While answering these queries, this work seeks to contribute towards the development of developing intelligent, adaptive, and efficient communication systems following the principles of brain-like computation.

## 2 | Literature Review

The physical layer of wireless communications networks is at the forefront of ensuring safe and efficient information transmission. With modern networks transitioning from 4G and 5G to the future potential of 6G, the physical layer is presented with increasingly sophisticated challenges. These are the high mobility users, fast changing channels, multipath fading, and hardware-induced nonlinearities. For instance, in mmWave high frequency bands, the signal is strongly receptive to channel distortion and obstacles, resulting in decreased reliability and performance [19]. Moreover, in MIMO-OFDM systems with a large number of antennas and subcarriers used for spectral efficiency gain, the issue of precise channel estimation and symbol detection is much more challenging due to the massive amount of parameters and real time requirements [20].

In the past, model based signal processing techniques such as zero forcing (ZF), MMSE detection, or LS estimation have been used to solve such problems. Although robust in controlled environments, they tend to be designed on the basis of idealized models that are not necessarily valid in actual conditions. Additionally, the majority of these techniques tend to demand perfect channel knowledge, which is not always known or time invariant under fast varying or nonlinear environments.

To counteract these limitations, machine learning (ML) methods have become a powerful set of competitors able to learn patterns directly

from data without relying on explicit mathematical definitions. ML algorithms are adaptive, nonlinear-resistant, and minimize the need for manual tuning. Of these methods, Reservoir Computing (RC) and its popular variant, the Echo State Network (ESN), is a light and computationally friendly algorithmic solution relative to deep learning methods such as LSTMs or Transformers [9, 21].

ESNs are especially well matched to time-series issues with their recurrent architecture and dynamic memory. Their ease of training learning the output weights only is appealing in computationally strapped applications, such as satellite receivers or edge devices. Wireless systems have had other applications of ESNs in symbol detection [22], channel equalization [23], and nonlinear distortion cancellation [24]. Their behavior on these operations has the potential to redefine physical-layer processing in forthcoming generations of communication systems.

A detailed overview of how Reservoir Computing, i.e., ESNs, have been used in wireless physical-layer challenges continues in this chapter. It starts with presenting the pivotal concepts of OFDM and MIMO technology to lay the groundwork. Thereafter, background theory of RC/ESNs and their advantages over traditional and other ML based approaches are described. The chapter also provides an overview of the latest uses of ESNs in terrestrial as well as non-terrestrial networks, including channel estimation, symbol detection, and equalization. It also presents gaps and future research directions concerning ESN deployment in hybrid 5G/6G communications systems.

## **2.1 Foundations of Modern Wireless Communication**

The foundation of modern wireless communication rests on two key technologies: Orthogonal Frequency Division Multiplexing (OFDM)

and Multiple-Input Multiple-Output (MIMO). Together, they have transformed how wireless systems achieve high-speed, reliable data transmission across various environments.

### 2.1.1 Orthogonal Frequency Division Multiplexing (OFDM)

OFDM is a digital modulation scheme where a broadband frequency channel is divided into multiple narrowband subcarriers. The entire subcarriers transmit a part of the total data in parallel, which reduces the symbol duration considerably. This is the technique of mitigating frequency-selective fading, a generic problem in wireless channels where different frequencies are subjected to different attenuations [20].

OFDM builds on Frequency Division Multiplexing Frequency Division Multiplexing (FDM), where distinct data streams are allocated to separate parallel frequency channels, separated by guard bands to minimize inter-channel interference. OFDM differs from traditional FDM in key ways: (1) employing multiple orthogonal subcarriers to convey the data stream; (2) ensuring subcarrier orthogonality; and (3) adding a guard interval to each symbol to reduce channel delay spread and inter-symbol interference.

To prevent subcarrier interference, OFDM makes use of a mathematical property known as orthogonality. The CP serves as a guard interval to remove inter-symbol interference that arises due to multipath propagation, where signals bounce off surfaces and reach the receiver with minor delays [19].

The accompanying figure illustrates OFDM's core concepts and the frequency-time domain relationship. In the frequency domain, adjacent subcarriers are modulated independently with complex data, followed by an inverse Fast Fourier Transform (IFFT) to generate the time-domain symbol.

Guard intervals are inserted between symbols to prevent inter-symbol

interference from multipath delays. Symbols are concatenated to form the OFDM burst, and at the receiver, a Fast Fourier Transform (FFT) recovers the original data bits.

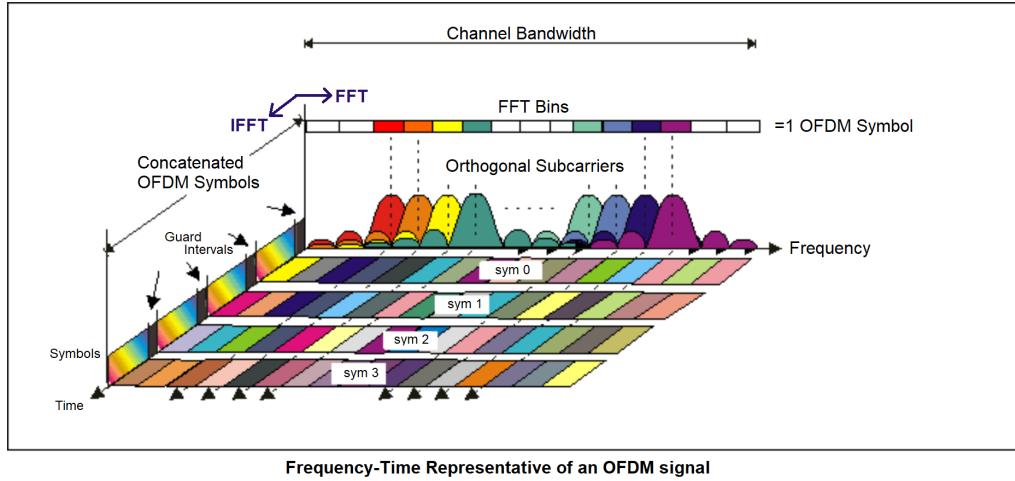


Figure 2.1: Combined time/frequency domain view of OFDM signal. Adapted from [2].

Due to its reliability and efficiency, OFDM has been widely used in wireless standards like Wi-Fi (IEEE 802.11), LTE, and 5G New Radio (NR).

### 2.1.2 Introduction to SISO, SIMO, and MIMO Systems

Wireless systems can be categorically described based on the number of transmitter antennas and receiver antennas. The three basic configurations are: Single-Input Single-Output (SISO), Single-Input Multiple-Output (SIMO), and Multiple-Input Multiple-Output (MIMO). Each of them has with it special characteristics that affect system capacity, reliability, and complexity.

**SISO:** The most primitive setup, one transmit and one receive antenna. Easy to deploy and measure, but not very capable of fighting multipath fading and not able to provide spatial diversity or multiplexing advantage. With simplicity, of course, aside, SISO is the reference point by which improved systems are judged.

**SIMO:** SIMO is an extension of SISO with multiple receive antennas

and a single transmit antenna. SIMO allows the exploitation of spatial diversity, enhancing receiver reliability with methods like Maximal Ratio Combining (MRC) or Selection Combining (SC). SIMO is applied when receiver performance enhancement at the cost of no additional transmitter complexity is required.

**MIMO:** MIMO systems employ multiple antennas at both receiver and transmitter ends, and they provide promise for radical data-rate increase and link reliability. Rather than regarding signal reflections and multipath as impairments, MIMO systems take advantage of them using sophisticated techniques like:

- **Spatial Multiplexing:** Allows parallel transmission of separate data streams and thus enhances spectral efficiency.
- **Diversity Techniques:** Repeat the same message over different paths to make the system more resistant to fading.
- **Beamforming:** Channels signal energy in certain spatial directions to offer enhanced signal quality and less interference.

To utilize MIMO to its full potential, precise CSI is needed. CSI offers insight into the propagation features of the channel, allowing the system to modify its transmission scheme in real time. Nevertheless, obtaining reliable CSI particularly in nonlinear or high-mobility channels—is one of the major hurdles in practical implementations [25, 26].

All three configurations SISO, SIMO, and MIMO are considered in this thesis within the framework of OFDM-based symbol detection issues. This tiered analysis enables comparative analysis and highlights the scalability of model proposals with learning under changing levels of system complexity.

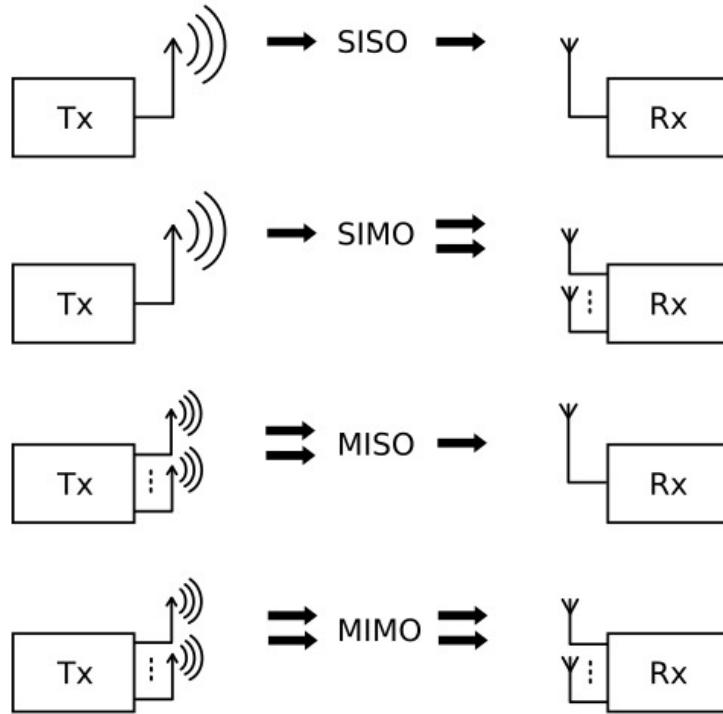


Figure 2.2: Illustration of SISO, SIMO, MISO, and MIMO antenna configurations.  
Source: Adapted from Wikimedia Commons [3].

### 2.1.3 MIMO-OFDM Systems

When MIMO and OFDM are together, they create a rich system which is employed in all the prominent contemporary wireless standards. OFDM supports frequency-selective fading, but MIMO builds capacity by spatial diversity and multiplexing [19]. Using both the systems also makes the system complex. Channel estimation among subcarriers and among various antennas, interference processing and hardware-imposed nonlinearities have to be handled by the receiver.

Conventional signal processing algorithms like Least Squares (LS), Zero-Forcing (ZF), and Linear Minimum Mean Square Error (LMMSE) estimators have been widely used for symbol detection and channel equalization in MIMO-OFDM systems over several decades. The conventional algorithms do show excellent performance degradation under rapidly changing channels or nonlinear degradations, particularly when perfect channel state information (CSI) is unavailable or is missing.

These issues have driven researchers to look for new approaches, and this is where **Reservoir Computing** and **Echo State Networks** appear. These techniques are themselves designed to address temporal dynamics and nonlinear relations, providing a different approach to channel estimation, symbol detection, and equalization in MIMO-OFDM systems.

## 2.2 Reservoir Computing and ESN Applications Physical Layer Processing

The physical layer in wireless communications systems provides the basis of coherent data transmission, such as the most basic operations including channel estimation, equalization, synchronization, and symbol detection. Due to the rising complication of modern communication environments such as rapid fading channels, high mobility, nonlinear hardware distortions, and erroneous CSI conventional signal processing methodologies are not accurate, flexible, and computationally complex.

RC, in this particular form of ESNs, has been an elegant light weight machine learning architecture to solve such problems. In contrast to conventional deep neural networks that involve extensive training and vanishing gradients, ESNs leverage static recurrent connections and learn only the output layer so that convergence is fast and temporal modeling is robust. Their capability to identify dynamic patterns positions them in an ideal position to be leveraged for any number of physical layer applications, from symbol detection and channel prediction through to beamforming assist and interference cancellation.

Recent investigations have proved the excellence of ESN-based models in some wireless system configurations, i.e., SIMO, SISO, and MIMO configurations, and in diverse propagation scenarios. Architectural

adaptations like multi layer ESNs , sliding window inputs, and frequency-domain implementations greatly enhanced performance in applications with data scarcity or hardware-imposed nonlinearities.

It presents an overview of the literature that currently addresses physical layer wireless communications based on RC and ESNs, pointing out their potential for resilient, efficient, and adaptive processing in future wireless systems.

### **2.2.1 Channel Estimation using ESNs**

Good data recovery in wireless communication begins with good channel estimation. Normal methods like LS or LMMSE are good under ideal situations but usually not effective under nonlinear or time-varying channels.

ESNs share the key strength of being able to deal with sequences effectively by extracting the short-term and long-term temporal dependences through their recurrent dynamics. The model is light and responsive as the readout layer is trained in isolation. It has been shown that ESNs can outperform traditional estimators in adverse conditions, e.g., non-Gaussian noise or rapidly varying fading channels [21, 27]. Their effectiveness makes them particularly useful in modern wireless scenarios, where pilot symbols are limited and channel feedback is delayed or expensive.

### **2.2.2 Symbol Detection**

Symbol detection in MIMO-OFDM receivers is yet another application area where ESNs excel. Classic approaches such as Zero Forcing (ZF), LMMSE, or maximum likelihood detection assume ideal channel knowledge and tend to require sophisticated computations. ESNs, on the contrary, can learn how to produce transmitted symbols from received signals directly without channel estimation explicitly.

In a single traditional paper, Mosleh et al.[22] demonstrated that ESN-based detectors exhibit competitive BER performance, particularly for fast-fading or even for unknown channels. Due to the reservoir's context memory across symbol time steps, it allows for enhanced immunity against sudden changes permitting low complexity and CSI free detection instead of traditional detectors.

### 2.2.3 Other Applications

Apart from channel estimation, detection, and equalization, ESNs also emerge to be useful in some novel applications of wireless signal processing:

- **Interference Mitigation:** The ability of ESNs to extract useful signal from interference via learning of temporal signatures useful in dense urban environments or overlapped transmission [28].
- **Dynamic Spectrum Allocation:** They may be employed to provide user traffic or channel availability predictions for enabling smarter and quicker resource allocation [29].
- **Antenna Selection and Beamforming:** Multi-antenna ESNs can discover spatial correlations and assist in selecting the best beams or the best set of antennas to achieve the best received signal quality [30].

These articles describe the ways in which RC and ESNs render wireless processing adaptive, low power, and real time potentially fulfilling the energy efficiency and latency requirements of modern wireless communication.

## 2.3 Physical Layer Processing through ML/DL Models

Though ESNs have demonstrated potential in handling dynamic and nonlinear channel conditions for MIMO OFDM systems, other ML

have also been widely studied for physical layer processing. These algorithms, i.e., Convolutional Neural Networks CNN, Recurrent Neural Networks RNN, Long Short Term Memory Long Short-Term Memory (LSTM) networks, Graph Neural Networks Graph Neural Network (GNN), and Transformers, possess respective strengths for different signal processing tasks. This section discusses their applications, benefits, and drawbacks over ESNs.

### **2.3.1 Constellation Identification and Modulation Classification**

The most pervasive physical layer use of ML models is modulation classification. CNN have been employed to identify modulation schemes by direct processing raw IQ samples with good success. Unlike traditional methods that include manual extraction of features, CNNs learn hierarchical features automatically from signal data and hence prove to be noise and distortion insensitive [31].

For instance, Giuliano and Innocenti [32] demonstrated that modulation classifiers based on CNN outperform traditional algorithms under low signal-to-noise ratio. In addition, the integration of CNNs with RNN enhances the classification accuracy under highly dynamic conditions because RNNs identify temporal dependencies in the modulation patterns.

### **2.3.2 Channel Estimation and Equalization**

Channel estimation is an important process in wireless systems, particularly in MIMO-OFDM systems where a precise channel state information (CSI) is required. ML based structures such as CNN and LSTM have been utilized for channel estimation and equalization because they are capable of capturing the intricate, nonlinear behavior of the channel.

CNN-based estimators learn antenna spatial correlation while LSTMs use temporal patterns to forecast channel fluctuation with time. These

models are more accurate compared to traditional models such as Least Squares (LS) and Linear Minimum Mean Square Error (LMMSE) in environments that experience rapid fading [33].

But one of their limitations is their training computational complexity that can be too large for real time use. Conversely, ESNs, due to their computationally inexpensive readout training, offer a less resource-hungry alternative when fast adaptation is needed.

### 2.3.3 Symbol Detection

RNN and LSTM have been employed intensively in symbol detection schemes for direct prediction of received symbols from the sequences of received signals. RNN and LSTM are optimized to capture time varying channel behavior, especially fading channels [34]. Though they are very accurate, there exists a need for massive training data and computational power, which restricts them to be deployed in low-power or resource-poor devices.

Transformers, which were recently used for symbol discovery, utilize self-attention to see long-distance dependencies without the vanishing gradient issue of RNN. Computationally costly but highly capable, they are not as real-time physical layer feasible as ESNs.

### 2.3.4 Spectrum Allocation and Interference Management

GNN are extremely popular because they can model complicated topological relations in wireless networks. In applications, for example, in interference management, GNNs represent interactions between spatially positioned antennas and users in a way that facilitates efficient resource allocation and interference avoidance [35].

Reinforcement Learning (RL) methods such as Deep Q-Networks (DQNs) have also been employed in dynamic spectrum allocation, where the model is trained to learn optimal usage of the spectrum

based on real-time reward signals. They are adaptive but take a large amount of exploration to converge and are therefore slow in heavily dynamic scenarios [36].

The growing use in wireless systems in the next generation, particularly for Sixth Generation (6G), has brought new challenges to physical layer signal processing, mainly because of the fast-changing channel conditions resulting from high mobility, Doppler shift, and long propagation delays [37]. Traditional equalization and channel estimation methods, which typically rely on quasi-static or slowly changing channels, are faced with such dynamic conditions. Under such conditions, ESNs could have much to offer with their temporal memory and adaptability at minimal additional computation expense.

### 2.3.5 Nonlinear Channel Equalization in LEO Satellites

Reservoir-based systems, in particular ESNs, have been found effective in Non-Terrestrial Network (NTN) systems that have nonlinear and time-varying channels. Bauduin et al. [23] proved that an glsen-driven equalizer could process nonlinear satellite channels with similar performance as conventional Volterra-based equalizers, even in noisy conditions. Their results highlight the ESN's capability to retain sequence memory as it learned instantaneous channel distortions a definite advantage in Low Earth Orbit (LEO) satellite links where Doppler shifts can amount to tens of kilohertz [38].

### 2.3.6 Adaptive Modulation and SNR Estimation

Another real application of ESNs in satellite communications was given by Sun et al. [24], wherein the authors employed ESNs in SNR estimation for ka band satellite systems. As opposed to traditional estimators, which are based on pre specified models and assumptions, the ESN based methodology learned signal to noise features from pilot data end to end. This allowed for better assistance with adaptive

modulation schemes typical in NTN systems where channel quality fluctuates tremendously over brief periods of time.

### 2.3.7 Real Time Adaptation in Dynamic Channels

Other recent research by Steck et al. [38] investigated adaptive equalization using ESNs in high-speed LEO satellite channels. Their research was on a LEO downlink with Doppler shifts up to 30 and bursty OFDM reception. The ESN could adapt to variations in the channel in real-time, as evidenced by a BER below  $10^{-3}$  in different configurations. These results verify that ESNs are able to retain performance over a broad Doppler range without retraining, as necessary for low-latency services over satellite-to-ground handovers.

**Challenges of Unified ESN Parameterization** While valuable, current application of ESNs to NTN systems entails scenario dependent hyperparameter adjustments such as reservoir size, spectral radius, and input scaling. This inhibits their extent of generalizability. No prior research has documented a specific ESN configuration that behaves in a consistent way over the entire Doppler and delay spread ranges encountered in hybrid Terrestrial Network (TN)/NTN environments, ranging from sub 1 in ground-level urban cells to  $>30$  in LEO passes [39]. Fulfilling this requirement is essential in deploying receivers based on ESN in commercial class heterogeneous networks.

While CNN, RNN, LSTM, GNN, and Transformers are likely to be physical layer processing candidates, they come with monumental computational requirements, especially training. They have high overhead of memory and latency arising from their recurrent nature and the attention mechanism. On the other hand, ESNs have minimal training requirements as they have a fixed reservoir and therefore are extremely resource frugal for real-time processing.

Apart from that, ESNs naturally deal with temporal dynamics whereas

CNNs and GNNs are better used for spatial pattern recognition. Thus, the hybrid approaches that integrate the temporal learning of ESNs with the spatial feature learning of CNNs can offer an optimally balanced solution for intricate wireless environments.

## 2.4 Performance Evaluations and Comparisons

Comparison of the performance of ESN based solutions in wireless communication systems must be performed cautiously by comparing them with conventional signal processing techniques and other ML-based solutions. The best performance metrics that can be used are BER, estimation accuracy, inference time, and complexity. Previous research shows that RC topologies especially those that use ESNs have been able to bring outstanding improvements under non linear conditions at a low implementation cost.

### 2.4.1 Comparison ESN with other ML models

Traditional techniques like ZF, LMMSE, and LS estimators are familiar in MIMO OFDM systems but are typically ideal channel assumption-based. If they are not tested under near-real conditions like with the occurrence of nonlinear distortion, restricted training symbols, or changing channel behavior they typically experience reduced accuracy and increasing BER. Conversely, ESN-based techniques enjoy the advantage of their recurrent dynamics, which enable them to learn and generalize on time-varying input patterns. As an instance, experimental performance results of [22] revealed that an ESN based symbol detector outperforms LMMSE detectors based on BER for non stationary MIMO channels. Similarly, [40] showed that pilot-aided RC significantly reduces BER OFDM systems compared to traditional Zero Forcing Equalization (ZFE), even with extremely severe nonlinearity due to LED saturation.

Machine learning (ML) and deep learning (DL) models are now a necessary component of physical layer processing in wireless communication systems of today. Along with ESNs, numerous other ML/ DL models such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory Networks (LSTM)s, and Transformers have been employed for tasks such as channel estimation, symbol detection, modulation classification, and equalization.

Although ESNs are computationally light and capable of real-time adaptation because of the light weight training process, there are some other ML/Deep Learning (DL) models that have certain other advantages. CNNs, for example, excel a lot at extracting spatial features from high level signal data and are therefore extremely helpful in modulation classification and signal denoising. RNNs and LSTMs excel highly at learning temporal dependencies, which is highly important in channels that have dynamic fading. Transformers, built on self-attention mechanisms, have been able to handle long-range dependencies in sequential data.

Even with these advantages, ESNs continue to outperform them under some situations, especially where low latency and real time processing are paramount, for example, high mobility MIMO OFDM systems. In contrast, CNNs and RNNs, though being more accurate, tend to demand long training and more computational resources. Transformers, though very efficient, are less ideal for edge computing because of their model complexity.

The quantitative analysis of the comparison among these models shows that although CNNs and LSTMs perform better in modulation classification and temporal data modeling, ESNs are the most hardware-efficient solution for real-time usage, particularly in situations needing swift adjustment and thin deployment.

Table 2.1: Comparison of ESNs and Other ML/DL Models in Physical Layer Tasks

Model	Strengths	Weaknesses	Applications	References
ESN	Low latency, real-time processing	Limited long-term memory	Symbol detection, equalization	[21], [22]
CNN	High accuracy in feature extraction	High computational cost	Modulation classification	[31]
RNN	Captures temporal dependencies	Prone to vanishing gradient	Time-varying channel estimation	[40]
LSTM	Long-term memory retention	Training complexity	Sequential data modeling	[27]

#### 2.4.2 Hardware Efficiency

The other significant aspect of performance assessment is hardware realization. Research has demonstrated that deterministic RC models such as the Simple Cycle Reservoir (SCR) and Cycle Reservoir with Jumps (CRJ) allow for efficient hardware mapping onto platforms such as FPGAs. For instance, Zhang et al.[39] recorded a 65 percent reduction in DSP slice utilization when implementing a fixed deterministic reservoir versus a sparse random ESN.

This extensive use of RC based circuits results in not only computational efficiency but also power awareness a vital attribute for edge devices, mobile receivers, and NTN transceivers with constrained memory and power.

Table 2.2 summarizes some of the selected results from the literature that reflect upon the performance of RC ESN-based methods compared to baseline baselines.

Table 2.2: Summary of RC/ESN Performance in OFDM-MIMO Systems

Application	RC/ESN Variant	Benchmark Method	Performance Improvement
Symbol Detection	FPGA-based ESN	LMMSE	Lower BER in dynamic MIMO channels [22]
Nonlinear Equalization (VLC-OFDM)	PA-RC	ZFE	Reduced BER under LED nonlinearity [40]
Channel Estimation	ESN	LS, LMMSE	Higher accuracy under low pilot density [27]
Real-Time Deployment	SCR/CRJ on FPGA	Random ESN	65% DSP reduction and real-time latency [39]

## 2.5 Synthesis of Research Gaps

Although the exciting potential of RC and, to an even larger extent, ESNs extends to a broad spectrum of wireless communications applications, numerous research gaps have yet to be filled in the interest of enabling the realistic implementation of ESN based architectures for next-generation TN and NTN systems, namely challenges associated with performance and resource limitations of 6G networks.

**Lack of Unifying Reservoir Configuration** Most of the current work tunes ESN hyperparameters like reservoir size, spectral radius, and input scaling on a scenario-by-scenario basis. For instance, Doppler shifts for LEO links might have a totally different setting compared to ground terrestrial small-cell networks below 1 [38, 39]. No current reservoir configuration or topology is good in overall environments. This limits reusability of ESN across heterogeneous networks and restricts scalability of ESN for real-world deployment.

**Integration with Cross-Layer Features** Most of the ESN deployments are directed towards solely physical-layer activities like detection and equalization. The next-generation systems, however, need closer integration across protocol layers. Pilot design adaptation, beamforming, and power amplifier (PA) pre distortion, for instance, may be aided by predictive modeling with ESN.

**Limited Dataset and Training Diversity** The training data employed in ESN research are often restricted to fixed modulation strategies or constant channel models. This prevents the generalizability ability of ESN receivers in realistic scenarios involving bursty traffic, channel faults, or hostile environments such as jamming. Second, the majority of ESNs have been offline trained with constant duration sequences, which can fail to capture real-time operation constraints in modern systems under continuous learning or online adaptation.

**Shortcoming of Long Term Stability Analysis** It is not common to come across studies related to the long term stability and reliability of ESN under sustained use or varied conditions. The fading memory aspect of ESN, although well suited for temporal dependency learning, causes state drift in the long term. This has the potential to invoke impaired detection performance or misadaptation on prolonged operation without retraining on a timed schedule. Robustness methods like regularization, dropout, or adaptive forgetting factors are not thoroughly researched in this regard [9].

### 2.5.1 Summary of Research Gaps

To bridge these gaps and fully harness the potential of ESNs in next-generation communication systems, future work should aim to:

- Develop unified ESN implementations generalizing over Doppler-delay profiles.
- Explore ESN based co-optimization strategies involving pilot

reuse, beamforming correction.

- Generate diversified, dynamic datasets characteristic of real 6G applications for resilient training.

Addressing these issues will be crucial to realizing the vision of RC enabled, low latency, and adaptive physical-layer components for resilient architectures.

## 2.6 Research Direction and Implementation Scope

In response to the identified research gaps concerning the practical applicability of Echo State Networks (ESNs) in next generation wireless systems, this thesis undertakes a targeted implementation and evaluation of an ESN based receiver for physical-layer symbol detection. Specifically, the architecture proposed in "*Brain Inspired Wireless Communications: Where Reservoir Computing Meets MIMO-OFDM*" [5] serves as the foundation for the experimental framework.

To thoroughly assess the flexibility and robustness of the ESN model, the implementation was extended beyond the original MIMO-OFDM setting. The receiver was evaluated under multiple wireless configurations, including single input single output (SISO), single input multiple output (SIMO), and multiple input multiple output (MIMO) systems. Furthermore, the system was subjected to a variety of realistic channel conditions additive white Gaussian noise (AWGN), the 3GPP clustered delay line (CDL) model, and a relay feeding channel scenario to examine its performance under different levels of fading, mobility, and nonlinearity.

In addition to the single layer ESN structure, a multi layer deep ESN architecture was explored. This variant extends the temporal modeling capacity of the network by stacking multiple reservoir layers, thus

enhancing the system's ability to capture complex time dependent features of the received signal.

By implementing and evaluating ESNs across diverse scenarios, this research aims to bridge the gap between theoretical potential and practical application in ESN based physical layer receivers. The findings from these implementations guide the system modeling choices, experimental design, and evaluation criteria presented in the following chapter.

### **3 | Echo State Networks: Architecture, Analysis, and Application in OFDM Systems**

Wireless communication is even more commonly used today generations from 4G, through 5G and on into the anticipation of 6G are increasing in complexity at the physical layer. All of these advancements have posed new challenges, such as dynamic channel conditions, user mobility, and non-linear effects due to the hardware (as outlined in the previous chapter). Such conditions, encountered more and more frequently in real-world deployments, are extremely difficult to address correctly while detecting and decoding the signal reliably.

Chapter 2 has already discussed about an extensive analysis of the standard and machine learning-based practices for physical-layer processing. The theoretical simplification of traditional signal processing methods, such as Zero-Forcing and Minimum Mean Square Error (MMSE) detectors have made them popular during the first years. However, their actual performance is typically subject to degradation in real-world scenarios especially for the case of MIMO-OFDM systems under harsh fading and interference conditions or using non-ideal hardware components. In more recent works, the use of deep learning models like CNN and RNN has been investigated as a replacement with comparable results on certain tasks.

In light of these challenges, Reservoir Computing and more specifically Echo State Networks (ESN) have shown great promise. The literature that was reviewed in Chapter 2 pointed out several strengths of ESN, which include its capacity to model temporal dependencies efficiently and with their fast, lightweight training process, which makes them suitable for use in dynamic wireless channels. In a range of tasks, including symbol detection and channel estimation, ESNs have been shown to perform as well as or better than more complex models but with lower computational requirements and latencies. In the meantime, several questions remain to be answered in terms of how to configure, benchmark and deploy ESN-based architectures for next generation wireless systems.

The chapter builds on and directly relates to these insights into whose being retrieved, why and the research gaps. Here, we target at the introduction of the system model and measurement methodology which is employed in the context of ESN-based receivers for time-varying nonlinear MIMO-OFDM environments. The rest of this paper provides details of the ESN receiver design, comparative baseline models as well as sets the stage for complete simulation and evaluation framework. This work aims to bridge the gap between theoretical potential and practical performance of signal processing in modern wireless communication through a rigorous methodology that is both repeatable as well as reproducible, laying foundation for the subsequent analysis and results.

### **3.1 Reservoir Computing (RC) and Echo State Networks (ESNs)**

Reservoir Computing (RC) is a framework developed to simplify the training of recurrent neural networks for sequential data processing. Among various RC models, the ESNs has become the most influential

and widely used, particularly in fields such as time series forecasting, speech recognition, system identification, and wireless communications [5, 9, 21].

### ESN for Time Series Prediction(Mackey Glass chaotic time series)

Before deploying ESNs for symbol detection, we verify that the network can learn a nonlinear, long-memory sequence. We use the chaotic Mackey Glass (MG) process

$$\frac{dx(t)}{dt} = \beta \frac{x(t - \tau)}{1 + x(t - \tau)^n} - \gamma x(t),$$

with the standard parameters  $\tau=17$ ,  $\beta=0.2$ ,  $\gamma=0.1$ ,  $n=10$ . An ESN (pyESN) is trained in teacher-forced mode to predict  $x(t+1)$  from  $x(t)$ , and then run *free* (auto-regressively) for a long horizon without ground-truth input.

Table 3.1: Mackey–Glass ESN configuration (sanity check).

Item	Setting
Dataset	MG time series, $\tau=17$ (precomputed <code>mackey_glass_t17.npy</code> )
Train / horizon	2000 training samples; 2000-step free run (iterative one-step)
Inputs / outputs	One input (bias stream of ones), one output $x(t+1)$
ESN	<code>n_res=500</code> , <code>spectral_radius = 1.5</code> , <code>random_state=42</code>
Readout	Linear regression on reservoir states (ridge default in pyESN)
Metric	RMSE over the 2000-step free-run segment

**Result.** Figure 3.1 overlays the true MG signal and the ESN’s free running prediction after the training boundary (dotted line). The ESN tracks the chaotic path closely across the full 2000 step horizon, indicating that the reservoir has captured the sequence’s temporal dependencies and nonlinear dynamics. This demonstration supports our use of ESNs as demappers: the ability to model long memory and

nonlinearity in a lightweight recurrent structure is directly relevant to time domain equalization and symbol prediction.

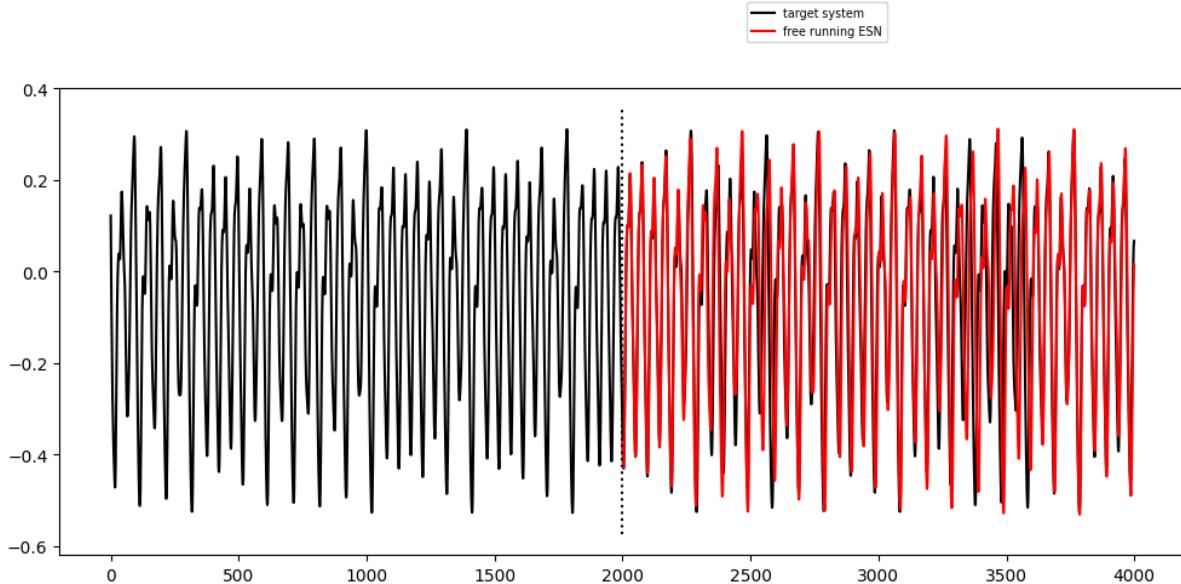


Figure 3.1: Mackey–Glass prediction with ESN. Black: target series; red: ESN free run after training (vertical dotted line marks the train/test split).

While reservoir computing is a general paradigm that includes a variety of models with fixed nonlinear reservoirs and trainable readouts, the Echo State Network (ESN) is a specific and mathematically well-defined realization of RC that employs a randomly connected recurrent neural network as the reservoir, along with the distinctive “echo state property” to ensure stability and reliable temporal processing.

An ESN consists of three main components: an input layer, a large and sparsely connected recurrent reservoir with fixed weights, and a linear readout layer. The input is projected into a high dimensional nonlinear state space by the reservoir, whose dynamics encode the temporal and nonlinear aspects of the input sequence. Only the output layer is trained typically using linear regression methods while the reservoir itself remains unchanged after random initialization. This architectural distinction drastically reduces training complexity and enables fast convergence, making ESNs highly suitable for real time and resource

constrained applications.

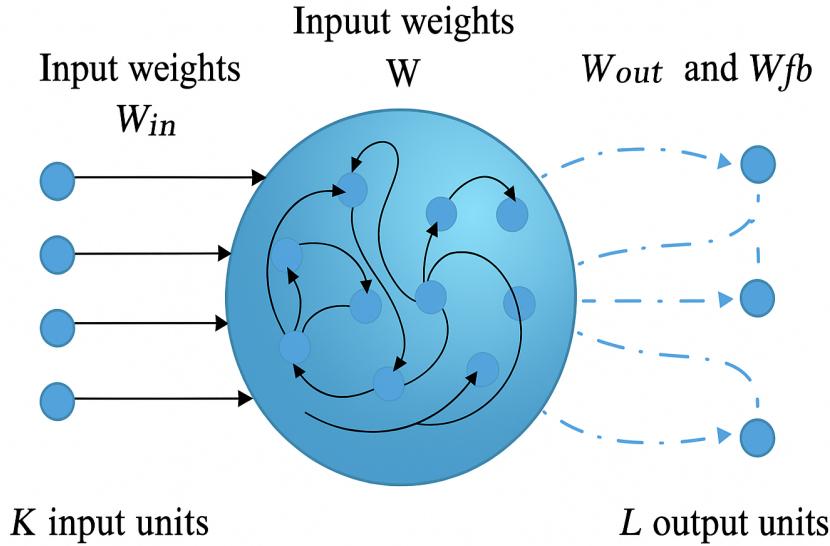


Figure 3.2: Echo State Network (ESN) architecture comprising input, dynamic reservoir, and trainable readout layers.

The fundamental properties that distinguish ESNs and make them effective for tasks like symbol detection in MIMO-OFDM systems include:

- **Fixed Nonlinear Reservoir:** The reservoir is a dynamic, nonlinear system with a fixed topology and weights. It acts as a Space-time related encoder, capturing the recent history of the input sequence.
- **Efficient Linear Readout Training:** Only the output weights are optimized, often by ridge regression or pseudoinverse computation, enabling rapid learning and minimal computational cost.
- **Temporal Processing and Memory:** The recurrent connections provide the network with a fading memory, allowing it to capture complex temporal dependencies in the input data.

While other RC variants exist, such as Liquid State Machines (LSMs), the ESN stands out due to its mathematical simplicity, robust stability

properties, and proven track record in engineering and wireless signal processing applications.

### **Mathematical Formulation of ESN**

The dynamics of an ESN can be described through its state update and output equations, as follows:

#### **State Update Equation:**

$$\mathbf{x}(n+1) = \tanh(\mathbf{W}_{in}\mathbf{u}(n+1) + \mathbf{W}\mathbf{x}(n) + \mathbf{W}_{fb}\mathbf{y}(n)) \quad (1)$$

Where:

- $\mathbf{u}(n) \in \mathbb{R}^K$ : Input vector at time step  $n$ .
- $\mathbf{x}(n) \in \mathbb{R}^N$ : Reservoir state vector.
- $\mathbf{y}(n) \in \mathbb{R}^L$ : Output vector from the previous time step.
- $\mathbf{W}_{in} \in \mathbb{R}^{N \times K}$ : Input weight matrix.
- $\mathbf{W} \in \mathbb{R}^{N \times N}$ : Recurrent weight matrix.
- $\mathbf{W}_{fb} \in \mathbb{R}^{N \times L}$ : Feedback weight matrix.
- $\tanh(\cdot)$ : Nonlinear activation function, applied element-wise.

#### **Output Equation:**

$$\mathbf{y}(n+1) = f_{out} \left( \mathbf{W}_{out} \cdot \begin{bmatrix} \mathbf{u}(n+1) \\ \mathbf{x}(n+1) \\ \mathbf{y}(n) \end{bmatrix} \right) \quad (2)$$

Where:

- $\mathbf{W}_{out} \in \mathbb{R}^{L \times (K+N+L)}$ : Trainable output weight matrix.
- $f_{out}(\cdot)$ : Output activation function, typically the identity function.

### Echo State Property (ESP)

A fundamental requirement for the ESN to function correctly is the *Echo State Property (ESP)*. This property ensures that the internal state of the reservoir becomes independent of its initial state after a sufficient number of time steps. Mathematically, the ESP is satisfied if the spectral radius of the reservoir weight matrix  $\mathbf{W}$  is less than unity:

$$\rho(\mathbf{W}) < 1 \quad (3)$$

This constraint guarantees asymptotic stability, such that the impact of initial conditions vanishes over time. The spectral radius  $\rho(\mathbf{W})$  must be carefully tuned depending on the application. Practically,  $\mathbf{W}$  is scaled after initialization as:

$$\mathbf{W} = \frac{\rho}{|\lambda_{\max}|} \cdot \mathbf{W}_0 \quad (4)$$

Where  $\mathbf{W}_0$  is the unscaled sparse random matrix, and  $\lambda_{\max}$  is its largest eigenvalue.

### Training Procedure of ESN

Among the inherent strengths of ESNs is that only the output weights  $\mathbf{W}_{out}$  need be trained, with the input and recurrent connections left as constants once initialized. Training procedure is typically to sample the reservoir states  $\mathbf{x}(n)$  over time for a given input sequence and then formulate a supervised regression problem to minimize the difference between predicted and target outputs.

Given a training set of input-output pairs  $\{\mathbf{u}(n), \mathbf{y}_{\text{target}}(n)\}$ , the readout weights  $\mathbf{W}_{out}$  are determined as the solution to the regularized least squares problem:

$$\mathbf{W}_{out} = \arg \min_{\mathbf{W}_{out}} \|\mathbf{Y}_{target} - \mathbf{X}\mathbf{W}_{out}^\top\|^2 + \lambda \|\mathbf{W}_{out}\|^2 \quad (5)$$

Where:

- $\mathbf{X}$  is the matrix of observed reservoir states (and optionally inputs and previous outputs),
- $\mathbf{Y}_{target}$  is the target output matrix,
- $\lambda$  is a regularization parameter to avoid overfitting.

The solution can be computed at an efficient rate utilizing ridge regression or pseudoinverse techniques. Because of the linearity of the output layer and fixed reservoir, training is much quicker than that of typical RNNs.

### 3.1.1 Deep Echo State Networks (DESN) for Symbol Detection

This thesis focuses mainly on the design and evaluation of a single layer ESN for symbol detection in MIMO OFDM systems, while also including a small exploratory study on DESN as an advanced extension. Following [4], the DESN stacks multiple reservoir layers so that different depths can capture temporal structure at different scales, which may ease the short memory issue observed in regular reservoirs. In our System model I setup, the DESN tended to outperform the plain ESN; at the same time it introduced more parameters, longer training, and, frankly, a loss of the straightforward simplicity that makes a basic ESN appealing. Given that trade off, I kept System model II focused on the single layer ESN to prioritize clarity of analysis, reproducibility, and manageable compute, while still allowing a fair comparison against the baselines.

A short description on here the output for each reservoir layer serves as the input for the next layer, and only readout layers are trained in the DESN. Speaking of learning long-range, nonlinear and more

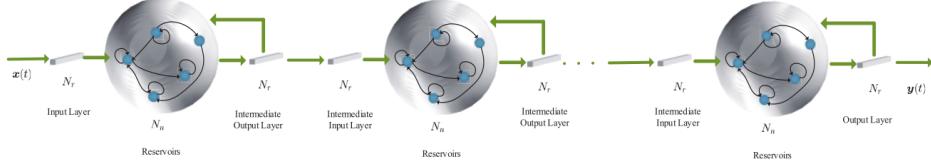


Figure 3.3: Deep Echo State Network (DESN) architecture comprising multiple stacked reservoirs with intermediate input/output layers. Each reservoir receives input from the previous layer, enabling hierarchical temporal modeling. Adapted from [4].

complex dependencies common in the difficult wireless channel environment where systems cannot always condition on all past symbols, this design allows for the network to capture temporal features at multiple levels of abstraction.

The training of the DESN is also computationally light, since only the output layers are updated, as in a conventional ESN. However, we need to tune the hyperparameters (number of layers, neurons per layer, reservoir connectivity), in order to avoid overfitting and for stability.

Refer to [4] for an extensive mathematical description and analysis.

## 3.2 System Architecture

The system architecture proposed is for solving the symbol detection problem of nonlinear SISO,SIMO,MIMO-OFDM systems. They are wellknown for their elevated spectral efficiency and link reliability because they use multiple antennas at both ends. However, with greater complexity comes the problem of accuracy in signal detection in a scenario with time-varying channel condition and nonlinear distortion. In this paper, we assess the performance of joint ESN-based neural demapping coupled with MIMO-OFDM receiver in a nonlinear, time-varying channel. We checked the performance on two variants of system itself as shown in Figure 3.4. The transmission side, common for both models, consists of a binary source → encoder → QAM

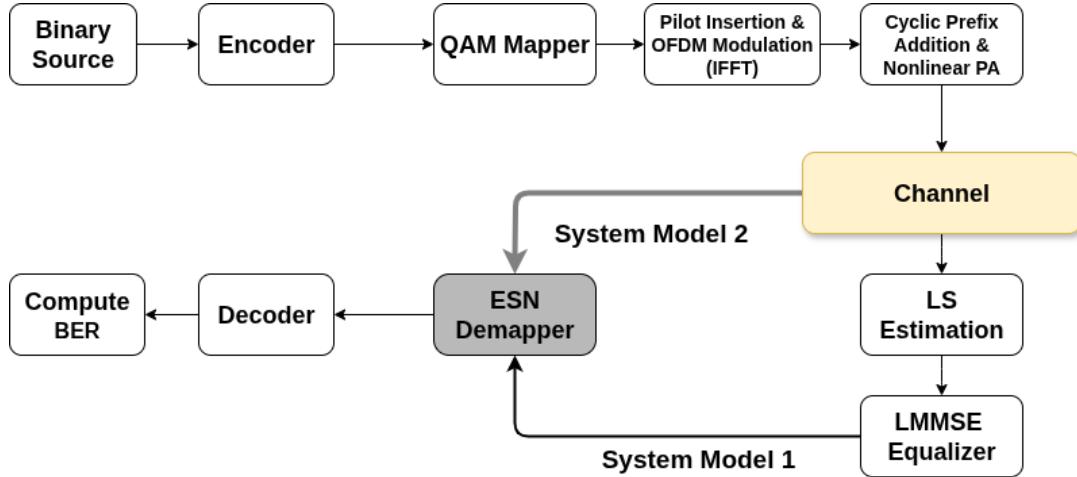


Figure 3.4: General Framework of the MIMO-OFDM System Architecture.

mapper and nonlinear time varying channel. The two variants of the system, as seen by the receiver are defined as:

### 3.2.1 System Model 1

The received signal is first subjected to traditional LS channel estimation followed by RLS-MSE equalization. This model provides an equivalent channel state information (CSI) and allows a fair comparison between classical and neural demapping. To this end, we consider a more realistic and challenging channel scenario where the Sionna framework is employed as the simulator with a CDL (Clustered Delay Line) channel together with  $4 \times 8$  MIMO antenna setup with complex 16 QAM modulation. Using the setup allows for comprehensive benchmarking of the performance of the hybrid receiver in challenging propagation conditions.

### 3.2.2 System Model 2

In System Model 2, the LS channel estimation and LMMSE equalisation blocks used in System Model 1 are *bypassed*. As depicted in Fig. 3.4, the ESN based detector operates on synchronised, CP added time domain receive samples and learns a direct mapping to the transmit domain OFDM waveform. The estimated waveform is then

transformed to the frequency domain via a DFT for symbol decisions. Hence, no explicit CSI estimation or linear equalisation is performed; instead, the detector leverages reservoir dynamics to accommodate channel dispersion and transmitter PA nonlinearity.

### ESN-Based Symbol Detection Scheme

In view of the nonlinear, time-varying weakness introduced by the transmitter PA and the frequency-selective, block-fading wireless channel, we adopt an ESN as a *time-domain* symbol detector that operates with minimal receiver-side preprocessing. Unlike the conventional pipeline—which relies on accurate CSI followed by LMMSE/ZF equalisation—the proposed detector does not perform explicit CSI estimation or linear equalisation. Instead, it learns a direct mapping from synchronised, CP-added receive samples to the transmit domain OFDM waveform, and decisions are then taken in the frequency domain after a DFT. This formulation follows the delay-aware reservoir approach in [5] and is instantiated here according to our implementation.

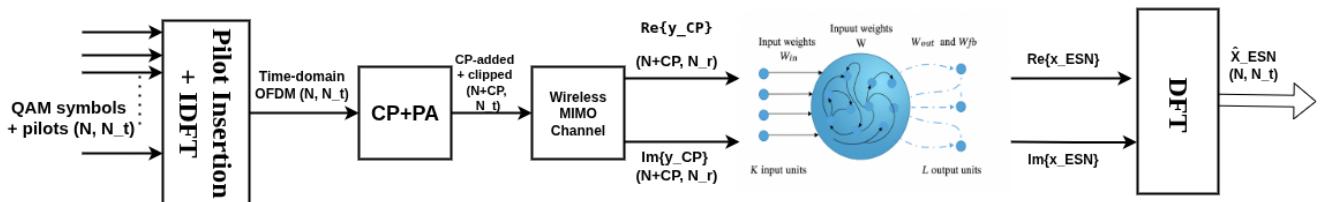


Figure 3.5: **System Model 2.** ESN-based time-domain detector with minimal front-end processing: pilot+IDFT → CP+PA → MIMO channel → Re/Im to ESN → DFT to obtain  $\hat{\mathbf{x}}_{\text{ESN}}$ .

### Step-by-step processing.

- 1) **Transmitter IFFT and scaling** Per OFDM symbol  $i$ , we use the non-unitary IFFT/FFT pairing implemented in code:

$$\mathbf{x}^{(i)} = N \text{IFFT}_N \{ \mathbf{X}^{(i)} \}, \quad \mathbf{X}^{(i)} = \frac{1}{N} \text{FFT}_N \{ \mathbf{x}^{(i)} \}, \quad (6)$$

with  $\mathbf{x}^{(i)} \in \mathbb{C}^{N \times N_t}$ , where  $N$  is the transform length and  $N_t$  the number of transmit branches. This convention is maintained consistently at the receiver.

**2) CP insertion and transmitter nonlinearity** A cyclic prefix of length  $N_{\text{CP}}$  is appended, producing  $\mathbf{x}_{\text{CP}}^{(i)} \in \mathbb{C}^{(N+N_{\text{CP}}) \times N_t}$ . The PA is modelled by a smooth-limiter (Rapp-type, shape  $p=1$ ):

$$\mathbf{x}_{\text{CP,NLD}}^{(i)} = g(\mathbf{x}_{\text{CP}}^{(i)}), \quad g(u) = \frac{u}{(1 + |u/A_{\text{clip}}|^{2p})^{1/(2p)}}, \quad A_{\text{clip}} = \sigma_x 10^{\text{ClipLevel}_{\text{dB}}/20}. \quad (7)$$

**3) Channel propagation and observation** Over a frequency-selective MIMO channel of memory  $L$  with AWGN, the CP-added observation is

$$\mathbf{y}_{\text{CP}}^{(i)} = \mathcal{H} \star \mathbf{x}_{\text{CP,NLD}}^{(i)} + \mathbf{n}^{(i)}, \quad (8)$$

where  $\mathcal{H}$  denotes the discrete-time convolution over the  $N_t \rightarrow N_r$  links. In our implementation,  $N_{\text{CP}}$  is chosen to cover the effective channel memory.

**4) ESN feature construction and targets** Because the reservoir nodes are real-valued, the ESN input at time  $n$  is formed by stacking real and imaginary parts across all  $N_r$  receive chains:

$$\mathbf{u}^{(i)}(n) = [\Re\{\mathbf{y}_{\text{CP}}^{(i)}(n, :)\}, \Im\{\mathbf{y}_{\text{CP}}^{(i)}(n, :)\}]^\top \in \mathbb{R}^{2N_r}. \quad (9)$$

The teaching signals are the real and imaginary parts of the transmit-domain time samples for each transmit branch; per-stream integer delays are used to accommodate channel memory and reservoir transient.

##### 5) ESN dynamics and readout estimation (teacher forcing; no feedback)

Training uses teacher forcing and *disables output feedback* ( $W_{\text{fb}}=0$ ),

aligned with the code:

$$\mathbf{x}(n+1) = f(\mathbf{W}_{\text{in}} \mathbf{u}(n+1) + \mathbf{W} \mathbf{x}(n)), \quad \hat{\mathbf{y}}(n) = \mathbf{W}_{\text{out}} [\mathbf{u}(n); \mathbf{x}(n); 1]. \quad (10)$$

Only  $\mathbf{W}_{\text{out}}$  is learned;  $\mathbf{W}_{\text{in}}$  and  $\mathbf{W}$  remain fixed. The readout is obtained by ridge regression (closed form):

$$\mathbf{W}_{\text{out}} = (\mathbf{S}^\top \mathbf{S} + \lambda \mathbf{I})^{-1} \mathbf{S}^\top \mathbf{T}, \quad (11)$$

where  $\mathbf{S}$  stacks rows  $[\mathbf{u} \ \mathbf{x} \ 1]$  over the training window and  $\mathbf{T}$  stacks the aligned targets. Setting  $\lambda=0$  recovers the Moore–Penrose solution.

**6) Delay/washout, and Fast Fourier Transform (FFT)-domain decisions** After a washout of  $n_{\text{forget}} = d_{\min} + N_{\text{CP}}$ , the next  $N$  time samples from each output stream are collected to form  $\hat{\mathbf{x}}_{\text{ESN}}^{(i)} \in \mathbb{C}^{N \times N_t}$ . Frequency-domain symbols for decisions are then obtained using the receive-side scaling paired with (6):

$$\tilde{\mathbf{X}}_{\text{ESN}}^{(i)} = \frac{1}{N} \text{FFT}_N \{ \hat{\mathbf{x}}_{\text{ESN}}^{(i)} \}. \quad (12)$$

These symbols feed the demapper to produce LLRs and the LDPC decoder for coded performance evaluation.

**7) Evaluation metrics and implementation notes** Unless otherwise stated, results are reported as *coded* BER (post-LDPC) at the demapper output. The pipeline in Fig. 3.5 bypasses all explicit channel-related blocks; learning is entirely data-driven in the time domain. The non-unitary transform convention (6) is used consistently at both ends of the link.

### Training of the ESN

The ESN is trained *offline* with teacher forcing on synchronised, CP-added receive sequences. Each sequence is partitioned into (i) an initial washout, (ii) a training window, and (iii) a held-out set for validation/testing. Only the linear readout is learned; the input map

and reservoir ( $\mathbf{W}_{\text{in}}, \mathbf{W}$ ) remain fixed, consistent with the RC framework.

**Data preparation and alignment** Per time index  $n$ , the input feature vector stacks real and imaginary parts across all  $N_r$  receive chains (cf. (9)):

$$\mathbf{u}(n) = [\Re\{\mathbf{y}_{\text{CP}}(n,:)\}, \Im\{\mathbf{y}_{\text{CP}}(n,:)\}]^\top \in \mathbb{R}^{2N_r}. \quad (13)$$

Teacher targets  $\mathbf{y}(n) \in \mathbb{R}^L$  are formed from the real/imag parts of the desired transmit-domain time samples, after applying the per-stream integer delays determined during synchronisation. The initial washout covers the minimum delay plus the CP:

$$n_{\text{forget}} = d_{\min} + N_{\text{CP}}.$$

**State update under teacher forcing (no output feedback)** Training uses teacher forcing and *disables* output feedback:

$$\mathbf{x}(n+1) = f(\mathbf{W}_{\text{in}} \mathbf{u}(n+1) + \mathbf{W} \mathbf{x}(n)), \quad \hat{\mathbf{y}}(n) = \mathbf{W}_{\text{out}} [\mathbf{u}(n); \mathbf{x}(n); 1], \quad (14)$$

where  $f(\cdot)$  acts elementwise and  $\mathbf{x}(n) \in \mathbb{R}^{N_{\text{res}}}$  is the reservoir state.

**Design and teacher matrices** Samples from the training window  $n = n_0, \dots, n_{\max}$  are stacked into

$$\mathbf{S} = \begin{bmatrix} \mathbf{u}(n_0)^\top & \mathbf{x}(n_0)^\top & 1 \\ \vdots & \vdots & \vdots \\ \mathbf{u}(n_{\max})^\top & \mathbf{x}(n_{\max})^\top & 1 \end{bmatrix} \in \mathbb{R}^{M \times D}, \quad \mathbf{T} = \begin{bmatrix} \mathbf{y}(n_0)^\top \\ \vdots \\ \mathbf{y}(n_{\max})^\top \end{bmatrix} \in \mathbb{R}^{M \times L}, \quad (15)$$

with  $M = n_{\max} - n_0 + 1$  and  $D = 2N_r + N_{\text{res}} + 1$ .

**Readout estimation (ridge regression)** The readout  $\mathbf{W}_{\text{out}} \in \mathbb{R}^{D \times L}$  is estimated in closed form via ridge regression to improve numerical

stability and mitigate overfitting:

$$\mathbf{W}_{\text{out}} = (\mathbf{S}^\top \mathbf{S} + \lambda \mathbf{I})^{-1} \mathbf{S}^\top \mathbf{T}, \quad (16)$$

where  $\lambda \geq 0$  controls the regularisation strength. Setting  $\lambda = 0$  gives the Moore–Penrose solution  $\mathbf{W}_{\text{out}} = \mathbf{S}^\dagger \mathbf{T}$ .

**Validation protocol and metrics** After fitting (16), the trained ESN is run on the held-out set. The reconstructed time-domain outputs (after per-stream delays and washout) are transformed using the receive-side scaling paired with the transmitter standard (cf. (6)):

$$\tilde{\mathbf{x}}_{\text{ESN}} = \frac{1}{N} \text{FFT}_N \{ \hat{\mathbf{x}}_{\text{ESN}} \}, \quad (17)$$

which feed the demapper to produce LLRs and the LDPC decoder.

**Computational remark** The dominant offline cost is solving the  $D \times D$  linear system in (16); with a direct solver, complexity scales as  $\mathcal{O}(D^3)$ . Runtime inference is lightweight, including only the reservoir update (14) and a dense readout evaluation per time step.

### Testing of the ESN

After training, the matrices  $\mathbf{W}_{\text{in}}$ ,  $\mathbf{W}$ , and  $\mathbf{W}_{\text{out}}$  are fixed (output feedback is *disabled*, i.e.,  $\mathbf{W}_{\text{fb}} = \mathbf{0}$ ). The ESN is then evaluated on a held-out sequence. The initial test state is warm-started either from the last training state or after a short on-sequence warm-up to mitigate transients. At each time step, features are formed from synchronised, CP-added samples as in (9); a linear readout produces the time-domain estimates; the reservoir state advances with the same nonlinearity used in training.

**Open-loop test recursion (no feedback)** Let  $\mathbf{u}(n) \in \mathbb{R}^{2N_r}$  be the input features at time  $n$ ,  $\hat{\mathbf{y}}(n) \in \mathbb{R}^{2N_t}$  the predicted real/imag streams, and

$\mathbf{x}(n) \in \mathbb{R}^{N_{\text{res}}}$  the reservoir state. Testing uses the same readout structure as in training (cf. (14), (16)) but without teacher signals or feedback:

$$\hat{\mathbf{y}}(n) = \mathbf{W}_{\text{out}} [\mathbf{u}(n); \mathbf{x}(n); 1], \quad (18)$$

$$\mathbf{x}(n+1) = f(\mathbf{W}_{\text{in}} \mathbf{u}(n+1) + \mathbf{W} \mathbf{x}(n)). \quad (19)$$

**Delay handling and FFT-domain decisions** Per-stream integer delays (estimated during training) are applied to align the two output streams (real/imag) of each transmit branch. After discarding the initial washout samples, the next  $N$  time samples are collected into  $\hat{\mathbf{x}}_{\text{ESN}} \in \mathbb{C}^{N \times N_t}$ . Frequency-domain symbols for decisions are then obtained using the receive-side scaling paired with the transmitter convention (cf. (6)):

$$\tilde{\mathbf{X}}_{\text{ESN}} = \frac{1}{N} \text{FFT}_N \{ \hat{\mathbf{x}}_{\text{ESN}} \}, \quad (20)$$

which feed the demapper to produce LLRs and the LDPC decoder.

**Performance measures** As a waveform-level check, we report the normalised root-mean-squared error (NRMSE) between the predicted and target time-domain sequences over a test window of length  $m$ :

$$\text{NRMSE} = \frac{\|\hat{\mathbf{y}} - \mathbf{y}\|_2}{\sqrt{m} \sigma_y}, \quad \sigma_y^2 = \text{Var}[\mathbf{y}], \quad (21)$$

where  $\hat{\mathbf{y}}(n)$  is obtained from (18). Link-level results are computed from the symbols in (20) as pre-decoder BER and coded BER (post-LDPC) on the same SNR grid used for baselines.

**Activation and scaling** The reservoir nonlinearity is  $\tanh(\cdot)$ , which is approximately linear near the origin and saturates away from it. Operation is kept in the near-linear regime via modest input scaling and per-frame normalisation of features. The readout activation is the identity, preserving the linear mapping learned during training.

In summary, testing mirrors training without teacher injection: initialise the state, run the open-loop recursion (18)–(19), apply delay alignment, convert to the frequency domain via (20), and evaluate both waveform error using (21) and link-level error after decoding.

### Tuning of the ESN

The ESN is tuned to balance accuracy, stability, and runtime under the minimal front-end pipeline of System Model 2. Choices follow our implementation practice and established guidance in RC. Inputs are constructed as in (9); state evolution and readout usage follow (14) and (16). In the result section 4.3, we also show how the parameters are selected with simulation results.

Let  $N_{\text{res}}$  denote the number of internal units. The readout feature width equals

$$D = 2N_r + N_{\text{res}} + 1, \quad (22)$$

corresponding to  $[\mathbf{u}; \mathbf{x}; 1]$  in (14). To avoid ill-conditioning and reduce overfitting, the number of training time steps  $M$  should exceed  $D$  with margin:

$$M \geq \gamma D, \quad \gamma > 1. \quad (23)$$

In practice,  $N_{\text{res}}$  is chosen so that (23) holds on all frames after the initial washout.

**Spectral radius (stability vs. memory)** The recurrent matrix  $\mathbf{W}$  is scaled to

$$\rho(\mathbf{W}) \approx \rho_0, \quad 0 < \rho_0 < 1, \quad (24)$$

to promote the echo-state property while preserving short-term memory. Values of  $\rho_0$  close to one (e.g., 0.9) provide stronger temporal integration without compromising stability.

**Input scaling (feedback disabled)** With  $\tanh(\cdot)$  nonlinearity in (14), operation near the linear region is maintained by small input scaling:

$$\mathbf{u}_{\text{scaled}}(n) = \alpha_{\text{in}} \mathbf{u}(n), \quad 0 < \alpha_{\text{in}} \ll 1. \quad (25)$$

Output feedback is *disabled* in our implementation ( $\mathbf{W}_{\text{fb}} = \mathbf{0}$ ), so no feedback scaling is applied.

**Putting it together** We select the reservoir size  $N_{\text{res}}$  such that the sample parameter ratio remains positive, i.e., the training window length satisfies  $M \geq \gamma D$  with margin (cf. (22)–(23)). The recurrent matrix is scaled to a spectral radius close to unity while strictly below one for stable memory,  $\rho(\mathbf{W}) \approx \rho_0 < 1$  (cf. (24)). Inputs are attenuated to keep the reservoir in the informative near-linear operating region of  $\tanh(\cdot)$ , using the factor  $\alpha_{\text{in}}$  in (25); output feedback is disabled in our implementation ( $\mathbf{W}_{\text{fb}} = \mathbf{0}$ ). A short washout of  $n_{\text{forget}} = d_{\min} + N_{\text{CP}}$  removes transients while preserving an effective sample count  $M_{\text{eff}} = M - n_{\text{forget}}$  that still meets (23). These choices follow established RC practice and are realised in our System Model 2 implementation.

To separate model capacity from channel complexity, we adopt a two-stage assessment:

1. **Baseline (noise-limited):** AWGN channel under SISO and SIMO configurations provides a controlled reference for time-domain learning and delay alignment.
2. **Impairment-rich (fading/nonlinearity):** A  $2 \times 2, 4 \times 8$  MIMO–OFDM link with relay-feeding and CDL and 4-/16-QAM examines scalability under frequency selectivity, inter-antenna coupling, and transmitter PA nonlinearity.

In both settings, the detector output feeds the standard demapper/decoder chain (LLR generation and LDPC decoding). Unless otherwise stated, performance is reported as coded BER

(post-LDPC); at the demapper output is additionally provided when informative. This design enables a like-for-like comparison of the ESN (i) as a *drop-in* substitute for a conventional demapper following front-end processing (System Model 1) and (ii) as part of an *end-to-end* receiver with minimal front-end processing (System Model 2).

### 3.3 Benefits of ESNs for Symbol Detection

Mapping received samples to the transmitted symbols in MIMO and OFDM receivers is rarely straightforward. Multipath and Doppler bend the waveform in time, front end nonlinearity introduces additional distortion, and AWGN sits on top. An ESN addresses this with a fixed recurrent reservoir that carries short memory, while a linear readout turns states into decisions.

#### 3.3.1 Key Advantages

- **Low training cost.** Only the linear readout is learned, typically by ordinary or ridge least squares. There is no back propagation through time. On short coherence blocks training may reduce to a single matrix solve.
- **Temporal modelling where it matters.** The reservoir keeps a rolling context over tens of samples, which often matches the scale of the CP and the effective channel memory. This tends to help when the channel is both time selective and frequency selective.
- **Tolerance to nonlinearity and noise.** Nonlinear state expansion can make mild transmitter PA compression and AWGN less damaging to separability before the readout. Gains appear most clearly when distortion is present but not extreme.
- **Real time inference.** State updates plus a dense readout are lightweight, so latency and energy use stay modest, which is appealing for embedded receivers.

- **Data efficiency.** Teacher forcing with a short washout exploits block fading stationarity. A few thousand labelled samples per operating point often suffice in practice.

### 3.3.2 Implementation Considerations

- **Feature design.** For time domain operation, concatenate real and imaginary parts from all receive chains at each time index. For frequency domain variants, use per subcarrier features. Normalising each frame, for example by its average magnitude, helps keep states in the informative region of the activation.
- **Delay alignment and washout.** Integer delays per stream align targets with predictions. An initial washout that covers the CP and the smallest expected delay removes transients. In our experiments a washout on the order of one CP length usually settles the state.
- **Reservoir hyperparameters.** Size, sparsity, input scaling, and spectral radius trade memory for stability. A spectral radius in the ballpark of 0.7 to 0.95 often gives expressive yet stable dynamics. Output feedback can remain off to simplify behaviour, as done in this work.
- **Training and metrics.** Train only the readout via least squares or ridge regression. Report symbol error rate and coded bit error rate after LDPC to allow fair comparison with classical receivers.
- **Compute aspects.** Readout estimation dominates training time and is an offline step. Runtime scales linearly with the number of time steps, and dataset generation benefits from based simulation, for example with Sionna.

### 3.3.3 Challenges and Limitations

- **Sensitivity to design choices.** Performance may drift with spectral radius, input scaling, sparsity, and size. Poor settings either saturate the states or starve the memory.
- **Random initialisation variance.** Different random reservoirs yield a spread in results. Fixing a good seed or averaging across a few seeds is a simple way to reduce risk.
- **Limited long term memory.** ESNs shine on short and medium contexts, which matches most OFDM links. Very long dependencies likely need architectural extensions.
- **Domain shift.** A strong mismatch between training and deployment, for example an abrupt change in SNR or channel statistics, can degrade accuracy. Light adaptation or periodic retraining often restores performance.

An ESN with a fixed reservoir and a trainable linear readout can serve as an efficient detector for MIMO and OFDM links. By carrying temporal context and tolerating moderate nonlinearity it offers a practical alternative when explicit CSI or linear equalisation struggle. These traits make ESNs promising for real time and energy conscious receivers in future Fifth Generation (5G) and 6G systems.

## 3.4 Baseline Models for Comparison

In order to evaluate the proposed ESN-based receiver for symbol detection in MIMO-OFDM systems, we need to assess its performance with respect to a number of well-known baseline models. Below is a summary that lists the classical signal processing methods alongside modern ML, which can be used as references for comparison.

### 3.4.1 Classical Signal Processing Methods

**Least Squares (LS) Channel Estimation:** As for the LS, it is a basic way to calculate the channel after receiving the pilot symbols which are used as training sequences alongside with data in transmitted signal. The strength of this approach resides on its computational simplicity and fast implementation. In particular, LSs estimators are not robust to noise and do not perform well when the SNR is low or in time-varying channels.

**Linear Minimum Mean Square Error (LMMSE) Equalization:**

LMMSE equalization uses the second order channel statistics to reduce the mean squared error between received and transmitted symbols. In overall error rate, LMMSE is superior over LS, particularly in noise and interference conditions; however, it requires for the noise variance to be known at the receiver and is computationally more expensive.

**Zero-Forcing (ZF) Equalization:** The main purpose of ZF equalizers is to diagonalize the channel matrix which results in complete cancellation of inter-symbol and inter-stream interference. Although very simple in concept, ZF tends to be particularly noise-sensitive, especially when the channel matrix is ill-conditioned, making it difficult for weak signals to be detected at an SNR level that is sufficiently usable.

**Conventional Demapper:** For all classical detection pipelines, a conventional symbol demapper is used following equalization. This block transforms equalized complex symbols into bit or symbol decisions using minimum-distance (maximum likelihood) rules according to the chosen modulation scheme (e.g., QAM).

These classical approaches define the standard “signal processing” baseline and are commonly used in practical wireless receivers. To this end, in the work at hand, we go on to juxtapose the ESN-based

demapper with a conventional equalizer to give an account of its respective merits and demerits under identical front-end processing.

### 3.4.2 Deep Learning–Based Baselines

In addition to the proposed reservoir computing receivers (ESN/DESN), we benchmark several widely used deep learning models as reference baselines for symbol detection in (nonlinear) MIMO–OFDM. The goal is not to exhaustively tune these models, but to position ESN/DESN against standard architectures with different inductive biases.

**Extreme Learning Machine (ELM).** ELM is a single hidden layer network whose input and hidden weights are fixed at random; only the linear readout is fitted in closed form. As a result, training is extremely fast and light weight. In this study, ELM serves as a minimal compute baseline that mirrors the ESN’s cheap readout fitting but without recurrent state: features per subcarrier (or per antenna subcarrier stack) are mapped directly to symbol logits or LLRs. This contrast isolates the contribution of temporal memory that ESN brings to time varying channels.

**Feedforward Neural Network (FNN/MLP).** A modest multi layer perceptron provides a strong static nonlinear benchmark. Inputs are vectorized IQ features formed from the equalized resource grid (e.g., concatenating antennas and subcarriers or using engineered statistics). Because each input is processed independently, the FNN tests whether a purely per-symbol universal approximator can close the gap to ESN under identical pilots and nonlinear hardware effects; prior work consistently reports that such generic DNNs benefit from larger training sets than are available in 4G/5G frames.

**Convolutional Neural Network (CNN).** A CNN treats the received IQ samples as a small “image” indexed by receive antenna and subcarrier

(optionally stacked over real/imaginary parts). Convolutions exploit local correlations across adjacent subcarriers and antennas that arise from filtering, OFDM structure, and spatial correlation. This baseline probes how far spatial frequency structure alone can carry symbol detection without explicit temporal state. CNN-based receivers of this type have been shown effective in OFDM detection and demapping pipelines (e.g., DeepRx and related designs), especially when CSI is imperfect; we follow that line while enforcing the limited-pilot regime used for ESN.

**Recurrent Neural Network (RNN: LSTM/GRU).** Gated RNNs propagate a hidden state across OFDM symbols and thus model Doppler induced time variation and residual inter symbol dynamics. In our setup, sequences of per symbol features are fed to a single or two layer LSTM/GRU, and time distributed outputs yield symbol logits/LLRs. This baseline represents the standard learnable-memory alternative to ESN; unlike ESN's fixed reservoir with closed form readout, RNNs require backpropagation through time and typically more data. The comparison highlights ESN's advantage in pilot-limited online learning while retaining temporal sensitivity.

### 3.4.3 Motivation and Comparison Strategy

The selection of these baselines are justified with the following reasons:

- To offer a balanced and in-depth comparison between classical and modern ways to page.
- To compare ESN performance not only with classical signal processing but also to well-established deep learning paradigms.
- Which aspects of ESNs are more unique (e.g. fast training, temporal modeling) and for which scenarios classical or deep learning may be a better alternative?

Results and detailed analyses are provided in the subsequent Results

and Discussion chapters, allowing the reader to draw meaningful conclusions about the suitability of ESNs and their competitive position within the wider landscape of symbol detection methods.

## 4 | Results and Analysis

In this chapter, the evaluation of the proposed receiver is carried out in stages. The aim is two part: first, to verify that an ESN can function as a practical demapper within a conventional OFDM–MIMO chain, and second, to examine how its performance changes when the model is fed signals taken closer to the direct channel output, where distortions are more pronounced.

We begin with a controlled end to end setup implemented in Sionna. In this baseline configuration, referred to as *System Model 1*, the standard baseband pipeline is kept intact (LDPC coding, OFDM modulation, pilot-assisted LS channel estimation, and LMMSE equalization). The only modification is that the conventional demapper is replaced with an ESN. This design isolates the demapping stage, making it possible to compare the ESN directly against classical references (LS, LMMSE with hard/soft demappers) as well as lightweight learning baselines such as ELM, FFNN, CNN, and RNN. To avoid overfitting the evaluation to specific conditions, all learning baselines are trained with fixed, moderate hyperparameters, without per SNR tuning. Labels are obtained directly from the coded bitstream, grouping  $m = 4$  bits per 16-QAM symbol. Alongside the post-LDPC BER, we also report uncoded Symbol Error Rate (SER) and include constellation plots for a visual check of decision regions.

After validating the ESN as a drop in demapper, we move to the more demanding *System Model 2*. Here, the ESN no longer sees equalized

constellation points but instead operates on signals that have undergone minimal receiver-side processing and are therefore much closer to the direct channel observations. This setup naturally exposes the model to residual frequency selectivity, inter-subcarrier phase drift, and the absence of explicit equalization. To separate model capacity issues from pure channel impairments, we gradually scale the scenario: starting with SISO transmission under AWGN, extending to SIMO, and finally returning to the full OFDM–MIMO case with frequency selectivity. Across these steps, stability is encouraged through simple techniques such as input normalization, short washout periods, and careful adjustment of recurrence or leak rate, while tracking how performance develops as realistic impairments are incrementally reintroduced.

For a fair comparison across both models, pilot density, SNR ranges, and computational budgets are aligned. Evaluation is based primarily on BER as functions of SNR, supplemented by runtime and an estimate of multiply–accumulate operations per frame as indicators of complexity.

## 4.1 Evaluation Strategy and First Pipeline

The evaluation is organised in stages. We begin with a controlled end-to-end receiver constructed with Sionna library to test whether an ESN can serve as a practical demapper. The idea is to first verify that the ESN works under standard OFDM–MIMO signal processing, before extending to more demanding cases where the ESN is exposed to signals taken closer to the direct channel output.

The first stage, referred to as **System Model 1**, corresponds to the receiver architecture already introduced in Chapter 3 (see Fig. 3.4). In this setup, all physical-layer blocks of the baseline chain remain unchanged (LDPC coding, OFDM modulation, pilot-aided LS channel

estimation, and LMMSE equalization), while the conventional demapper is replaced by an ESN. This allows a direct comparison of the ESN with both traditional rule-based demappers (LS, LMMSE) and standard machine learning models.

Table 4.1 summarizes the simulation settings applied throughout the experiments, ensuring that antenna configuration, modulation, pilot pattern, and SNR range remain consistent for fair comparisons.

Table 4.1: Sionna simulation parameters for System Model 1.

Parameter	Value
Modulation	16-QAM
OFDM symbols per frame	14
FFT size	76
Cyclic Prefix (CP) length	6 samples
LDPC code rate	1/2
Pilot pattern	Kronecker
Channel model	CDL-B (delay spread: 300 ns, speed: 10 m/s)
Antenna configuration	$4 \times 8$ (UT $\times$ BS)
SNR range	0–20 dB

The ESN is trained on real-valued features derived from the equalized constellation points. Labels are generated directly from the coded bitstream, grouping four bits into one of 16 classes for 16-QAM. The reservoir remains fixed, and only the readout layer is trained in a supervised manner. For comparison, we also evaluate other models under identical training conditions. Table 4.2 lists their configurations.

Table 4.2: Model configurations for System Model 1.

Model	Configuration
Traditional (LS/LMMSE)	Rule-based demapper
ESN	Reservoir size: 600; Spectral radius: 0.8; Dropout: 0.1; L2 regularization: $1 \times 10^{-4}$
DESN	Two ESN layers (600 units each); Spectral radius: 0.8; Dropout: 0.1; L2 regularization: $1 \times 10^{-4}$
FFNN	Dense(600, ReLU) $\rightarrow$ Dense(16)
CNN	Conv1D(128, kernel=1, ReLU) $\times$ 2 $\rightarrow$ Flatten $\rightarrow$ Dense(600, ReLU) $\rightarrow$ Dense(16)
RNN	SimpleRNN(600) $\rightarrow$ Dense(600, ReLU) $\rightarrow$ Dense(16)
ELM	Hidden layer: 600 units (fixed)

The evaluation metrics are:

- **BER** after LDPC decoding (main performance indicator).
- **Constellation plots** for visual inspection of learned decision regions.
- **Computational complexity**, measured via parameter counts, runtime per OFDM frame, and estimated multiply accumulate operations.

All models are trained in a supervised manner, with training and validation data derived from the encoder output. The ESN and ELM rely on fixed hidden representations (reservoir or hidden layer), training only the output layer via ridge regression, while the FFNN, CNN, and RNN are optimized end-to-end with cross-entropy loss.

#### 4.1.1 Results for System Model 1

Figure 4.1 reports the BER versus SNR. Classical demappers perform well at high SNR, while the DESN shows competitive results and consistently outperforms the single-layer ESN and the ELM. The CNN achieves moderate success, but the RNN and FFNN struggle to generalize under channel impairments.

Constellation diagrams in Fig. 4.2 show the decision regions of each model. The ESN and DESN produce stable clusters with some noise spreading, while the CNN and FFNN exhibit distortions or overlaps. The RNN fails to form consistent boundaries, which aligns with its weaker SER performance.

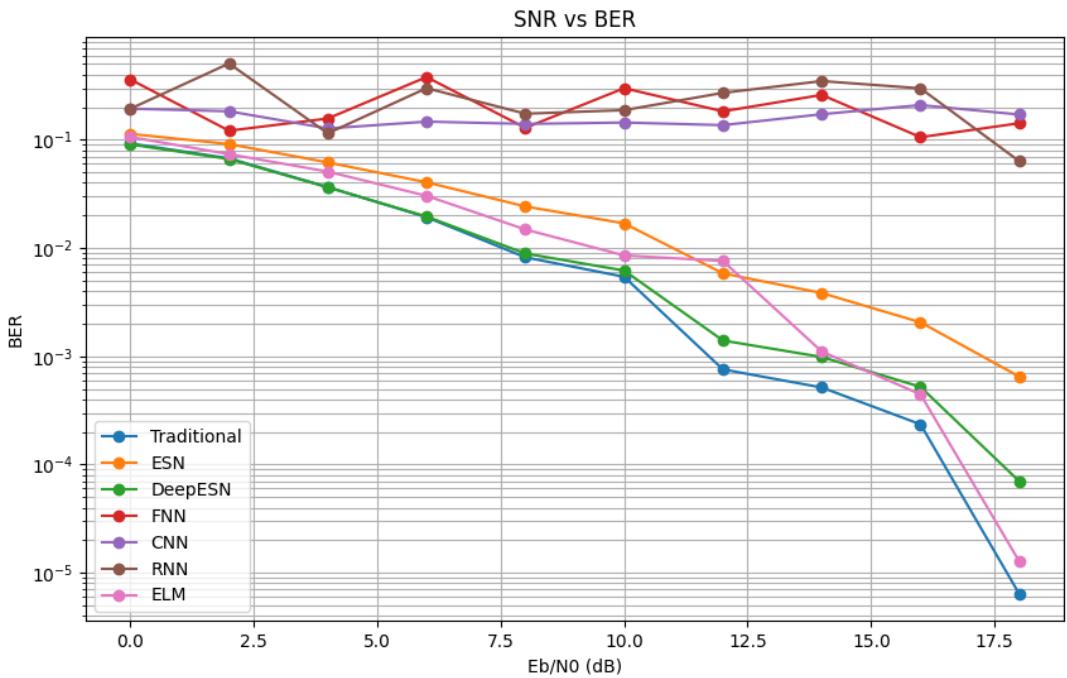


Figure 4.1: BER performance for different demappers in System Model 1.

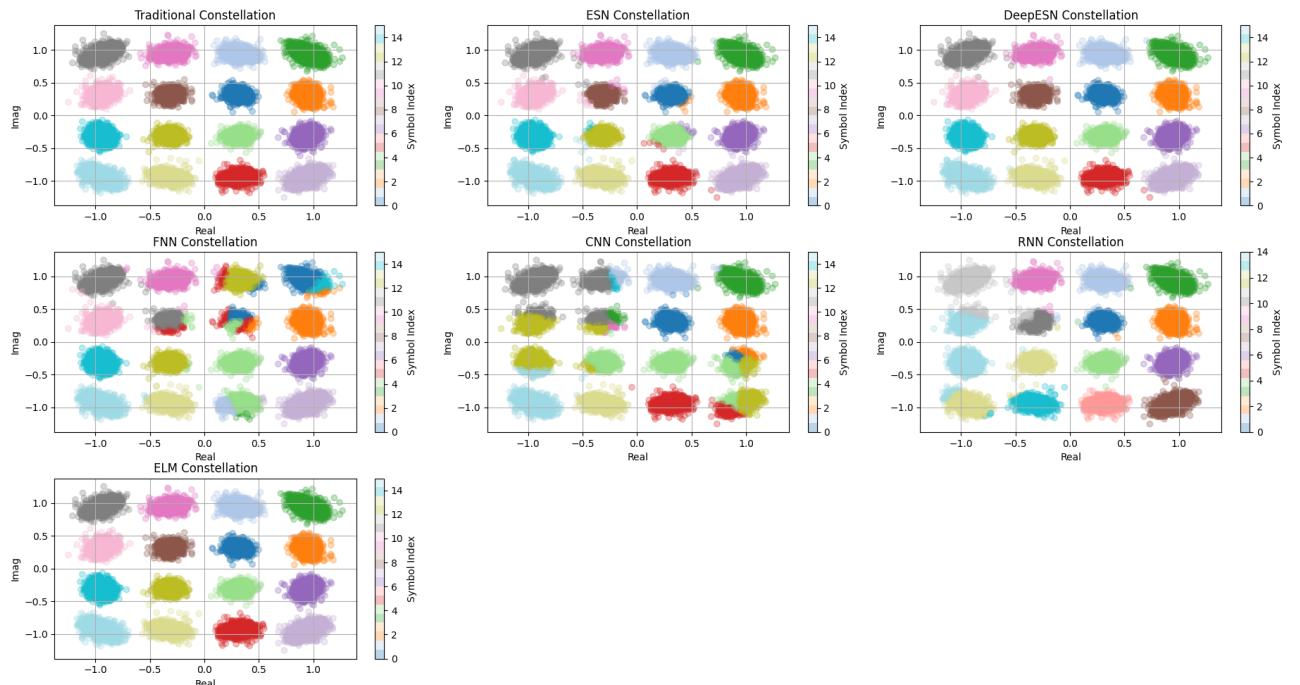


Figure 4.2: Constellation diagrams for different demappers in System Model 1, coloured by predicted symbol index.

## 4.2 System Model II: Results

This section presents the empirical outcomes of *System Model II*, following the staged evaluation pipeline defined in Section 3.2.2 (see Fig. 3.5) and (see Fig. 3.4).

To isolate the contribution of the RC/ESN demapper from channel and spatial confounders, we begin with a minimal AWGN configuration in SISO OFDM. This first stage verifies end to end correctness (symbol mapping, pilot/label alignment) and establishes baseline BER behaviour. We then progressively reintroduce realism frequency selectivity and Doppler in SISO, receive diversity in SIMO, and inter-stream interference in MIMO ( $2 \times 2$  and  $4 \times 8$ ) while keeping the training protocol, pilot pattern, and evaluation procedure fixed as specified in Section 3.2.2. Unless stated otherwise, comparisons include conventional LMMSE/ZF detectors and coded performance post-LDPC decoding, with SNR swept over the same range as in Chapter 3. The results are reported in the same order as the staged design, enabling a clear attribution of performance gains or degradations to specific physical layer factors (channel selectivity, pilot density, and spatial scaling) for each antenna configuration.

We will also discuss how the ESN parameters are selected for the simulation.

### 4.2.1 System Model II: Stage 1 OFDM SISO under AWGN

This first setup a clean AWGN reference for the proposed reservoir demapper before introducing frequency selectivity or spatial dimensions. after that we compare two learning receivers ESN (reservoir size matched against conventional baselines (Perfect CSI, LS, LMMSE)).

In order to test robustness, each learning model is evaluated in two

regimes: *matched SNR training* and *fixed SNR training* (trained once at 12 dB and tested across the full SNR sweep).

Table 4.3: System Model II – Stage 1 (short configuration).

Item	Setting
Scenario	SISO OFDM; AWGN ( $H_k = 1$ ); soft-PA (+3 dB, $p = 1$ )
OFDM parameters	$N = 512$ subcarriers; $N_{\text{cp}} = 0$
Modulation	4-QAM ( $m = 2$ ); pilot $m_{\text{pilot}} = 2$
$E_b/N_0$ grid	$\{0:3:30\}$ dB; $N_0 = 10^{-5}$
Data and pilots	400 data OFDM/SNR; 1 pilot/SNR; extra pilot at 12 dB (fixed train)
Learning models	ESN ( $n_{\text{res}} = 100$ , $\rho = 0.9$ ); delay search 0–2 (I/Q)
Baselines	Perfect CSI; LS; LMMSE
Metrics / outputs	Uncoded BER; $C = \log_2(1 + \text{SNR})$

A pilot OFDM symbol is generated at each SNR (and once at 12 dB for the fixed-train case). After IFFT and power scaling, a smooth limiter

Both learning models perform a small, discrete delay search (0–2 samples per I and Q) and select the best arrangement by normalised MSE against the pilot reference. For each SNR, 400 data OFDM symbols are transmitted; the ESN predicts *time-domain* clean signals which are FFT'd to symbols; conventional baselines are detected in the frequency domain. Hard decisions map to bits for BER.

## Results and Discussion

Figure 4.3 reports BER versus  $E_b/N_0$  for all methods. As expected under AWGN, Perfect-CSI approaches the analytical QPSK bound, and LS/LMMSE match with it given  $H_k \equiv 1$ . The ESN closely follows the reference at mid high SNR with a modest penalty at the lowest SNR points due to finite pilot symbols and the soft PA. With LS/LMMSE, basically, for practical SNR ranges.

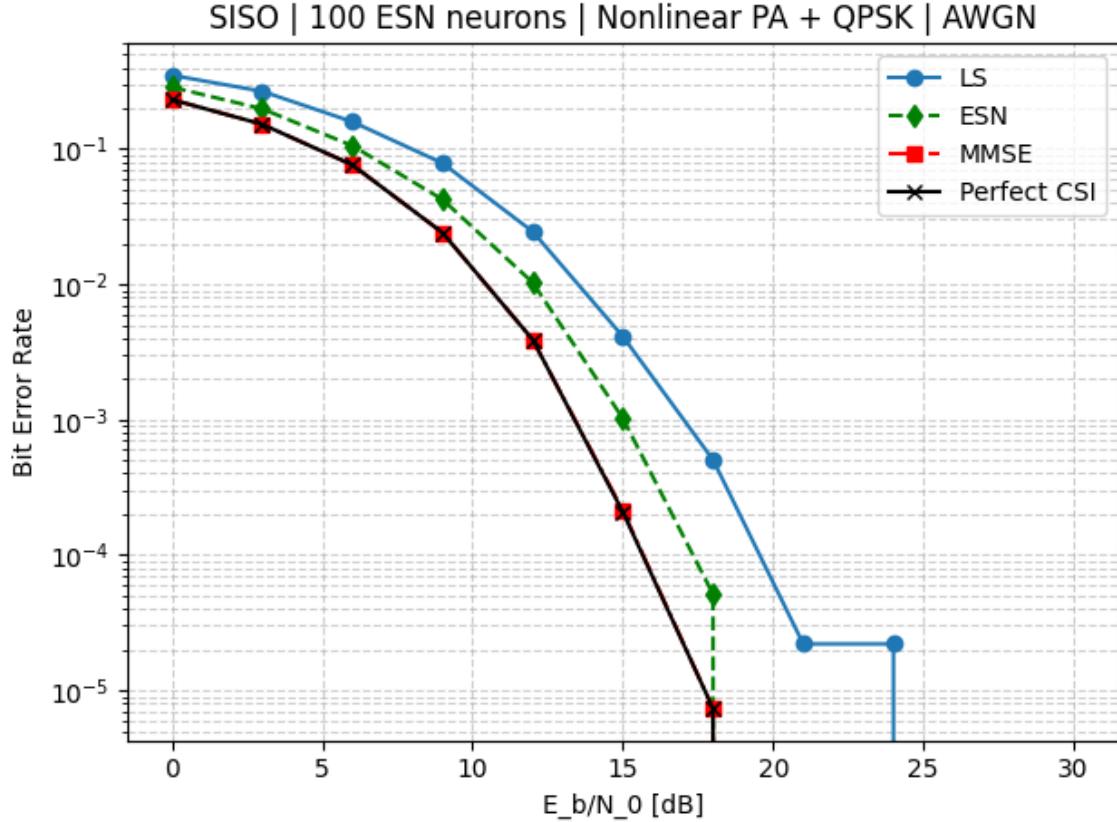


Figure 4.3: Uncoded BER vs.  $E_b/N_0$  for SISO 4-QAM under AWGN with soft PA: Perfect CSI, LS, LMMSE, ESN (matched and fixed-train@12 dB), and ELM (matched and fixed-train).

## Roadmap for the Remaining Outcome (System Model II)

With the ESN validated under AWGN, we now evaluate progressively richer propagation and spatial settings. Table 4.2.1 summarises the simulation setups used from this point on; all cases share the evaluation protocol from Section 3.2.2 (same pilot pattern, training recipe, and metrics).

<b>ID</b>	<b>Name</b>	<b>Antennas</b>	<b>Channel condition &amp; modulation</b>	<b>Model comparison</b>
<b>R1</b>	SISO block-fading	$1 \times 1$	Block-fading with $L=8$ taps (exp. PDP), $f_D=100$ Hz; soft-PA (+3 dB); 16-QAM; OFDM $N=256$ , $N_{cp}=7$ ; LDPC on.	LS, ZF, LMMSE
<b>R2</b>	SIMO block-fading	$1 \times 2$	Same channel per branch as R1 (block-constant across coherence); 16-QAM; OFDM $N=256$ , $N_{cp}=7$ ; LDPC as used.	LS, ZF, LMMSE
<b>R3</b>	MIMO block-fading	$2 \times 2$	Same PDP/Doppler per link as R1; inter-stream interference present; 16-QAM; OFDM $N=256$ , $N_{cp}=7$ ; LDPC as used.	LS, ZF, LMMSE
<b>R4</b>	MIMO block-fading (coded/uncoded)	$4 \times 8$	Higher-rank link; same PDP/Doppler per link as R1; 16-QAM; OFDM $N=256$ , $N_{cp}=7$ ; results pre-/post-LDPC.	LS, ZF, LMMSE
<b>R5</b>	MIMO model comparison	$2 \times 2$	Same as R3 (block-fading $L=8$ , $f_D=100$ Hz; 16-QAM; OFDM $N=256$ , $N_{cp}=7$ ); LDPC as used.	CNN, RNN, ELM, FFNN
<b>R6</b>	MIMO CDL(coded/uncoded)	$4 \times 8$	(CDL-B, $L=8$ , $f_D=100$ Hz; 16-QAM; OFDM $N=256$ , $N_{cp}=7$ ); LDPC as used.	MMSE

#### 4.2.2 SISO Block-Fading (16 QAM)Setup and Result

First, we considered the OFDM SISO setup.

Table 4.4: SISO configuration (OFDM, 16-QAM).

Item	Setting
Antennas	SISO $1 \times 1$
Channel	Block-fading, $L=8$ taps (exponential PDP), Doppler $f_D=100$ Hz; soft-PA nonlinearity
Modulation/OFDM	16-QAM; $N=256$ , $N_{cp}=7$
Coding	One systematic LDPC codeword per OFDM symbol, length $n = mN$ , with $m=4$
Receiver	ESN ( $n_{res}=300$ , spectral radius $\rho=0.9$ ); trained once per coherence block (short I/Q look-up)
Training regimes	Matched-SNR training; fixed training at 12 dB
Baselines	Perfect-CSI ZF, LS-ZF, MMSE
Metrics	Pre-decoder BER, post-LDPC BER (info bits), average per-subcarrier capacity from $\mathbf{H} = \text{diag}(H_k)$

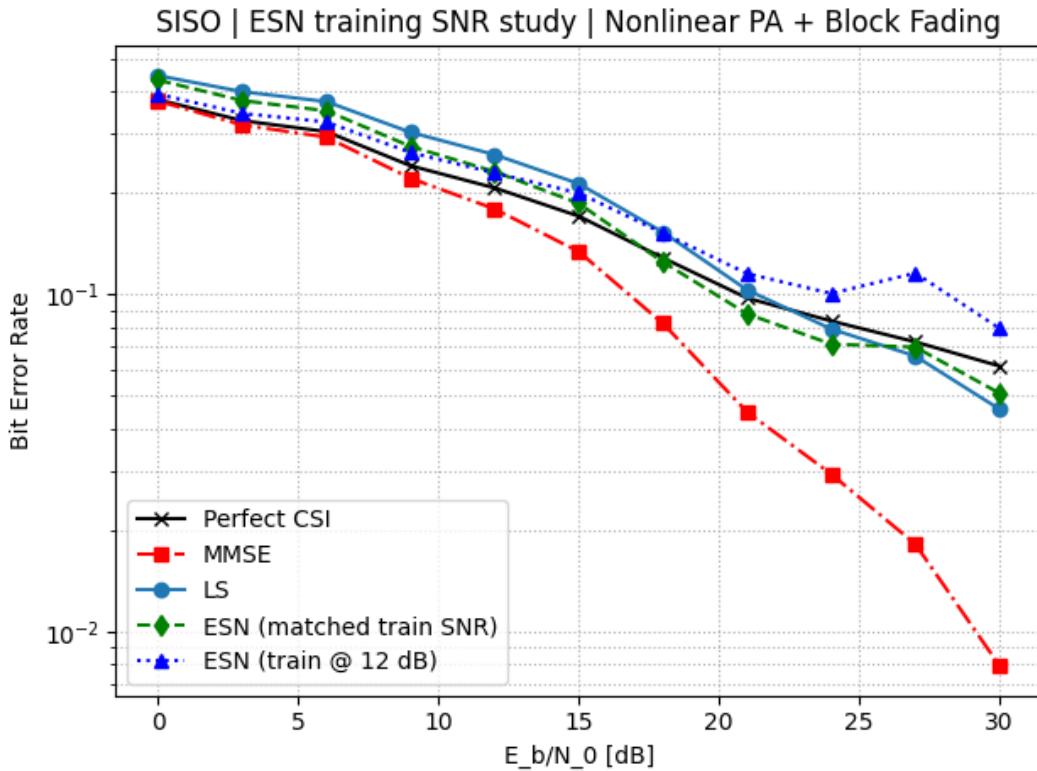


Figure 4.4: SISO, 16-QAM, block-fading with soft PA: uncoded BER vs.  $E_b/N_0$  for Perfect-CSI ZF, MMSE, LS-ZF, and ESN (matched and fixed-train at 12 dB).

**Results.** Figure 4.4 presents the training versus test SNR study. The *matched* ESN follows LS closely across the sweep; at very low SNR there is a small loss that likely

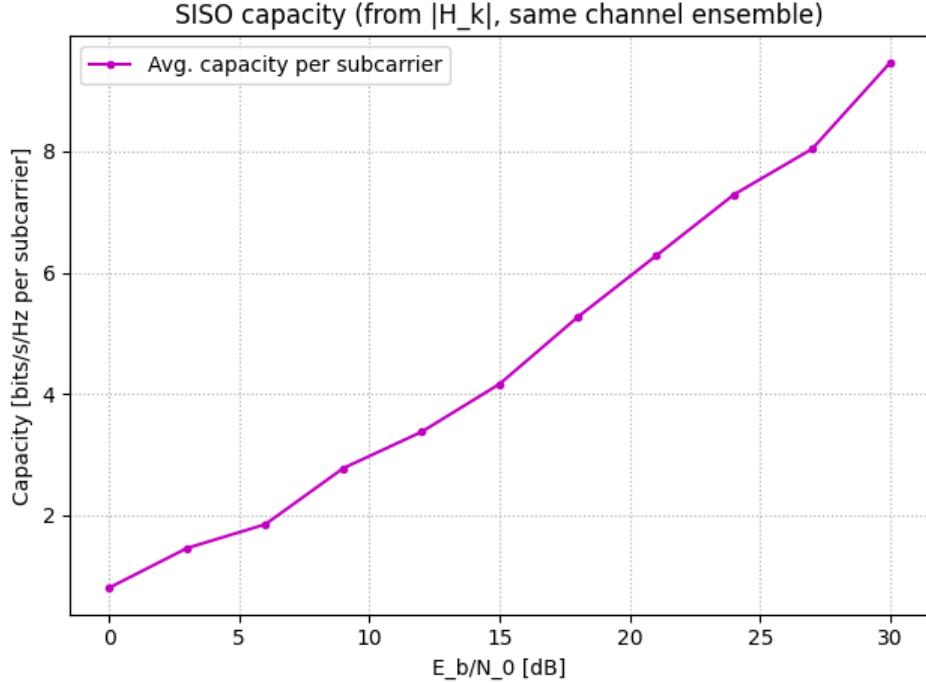


Figure 4.5: Average per-subcarrier capacity  $C = \mathbb{E}_k[\log_2(1 + \text{SNR}_k)]$  from the same  $\mathbf{H}$  ensemble for the SISO block-fading case.

comes from finite pilot energy and the soft PA. A model trained once at 12 dB degrades gradually. It stays competitive from 0 to about 21 dB and then begins to fall behind as SNR increases. As expected, MMSE leads in the high SNR range because it uses the noise variance explicitly, while Perfect CSI marks the oracle lower bound.

The capacity computed from the same channel ensemble (Fig. 4.5) rises monotonically with  $E_b/N_0$  and approaches just under 10 bits/s/Hz per subcarrier at 30 dB. This envelope helps explain the left shift of the BER curves as SNR grows. A slight wobble at the highest SNR values may reflect limited averaging or residual distortion from the soft PA.

1. A simple time domain inversion by the reservoir with a short delay lookup table is sufficient for  $L=8$  memory under mild nonlinearity. No back propagation through time was needed.
2. Training once at 12 dB appears to generalise across a wide SNR span, which is practical when frequent retraining is not feasible.
3. The remaining gap to MMSE at high SNR reflects the value of explicit noise aware equalisation. After LDPC decoding this gap usually narrows, although the extent depends on the code rate and the quality of the bit metrics.

The SISO block fading results indicate that the proposed ESN demapper handles frequency selectivity and mild nonlinearity well enough for our purposes and carries across SNR without constant retraining. We next move to SIMO to quantify diversity gains under the same protocol.

### 4.2.3 SIMO Block-Fading (16-QAM)

**Configuration** The SISO link is extended to a SIMO case with one transmit antenna and two receive antennas. A block-fading channel with exponential PDP, Doppler  $f_D=100$  Hz, and a soft PA (+3 dB) is applied. The ESN receiver uses a reservoir of 150 units with spectral radius 0.9, trained once per coherence block. Baselines rely on subcarrier-wise ZF detection with varying channel estimates (Perfect-CE, LS-CE, LMMSE-CE).

Table 4.5: R2 configuration (SIMO, 16-QAM).

Item	Setting
Antennas	SIMO $1 \times 2$ (stacked I/Q across two Rx branches)
Channel	Block-fading, $L=8$ taps (exp. PDP), Doppler $f_D=100$ Hz; soft-PA (+3 dB)
Modulation/OFDM	16-QAM; $N=512$ , $N_{cp}=7$
Receiver	ESN ( $n_{res}=150$ , spectral radius 0.9); matched-SNR training / fixed-training at 12 dB
Detectors	Subcarrier-wise ZF with {Perfect-CE, LS-CE, LMMSE-CE}
Metrics	Uncoded BER vs. $E_b/N_0$ ; capacity from $\ \mathbf{h}_k\ _2^2$

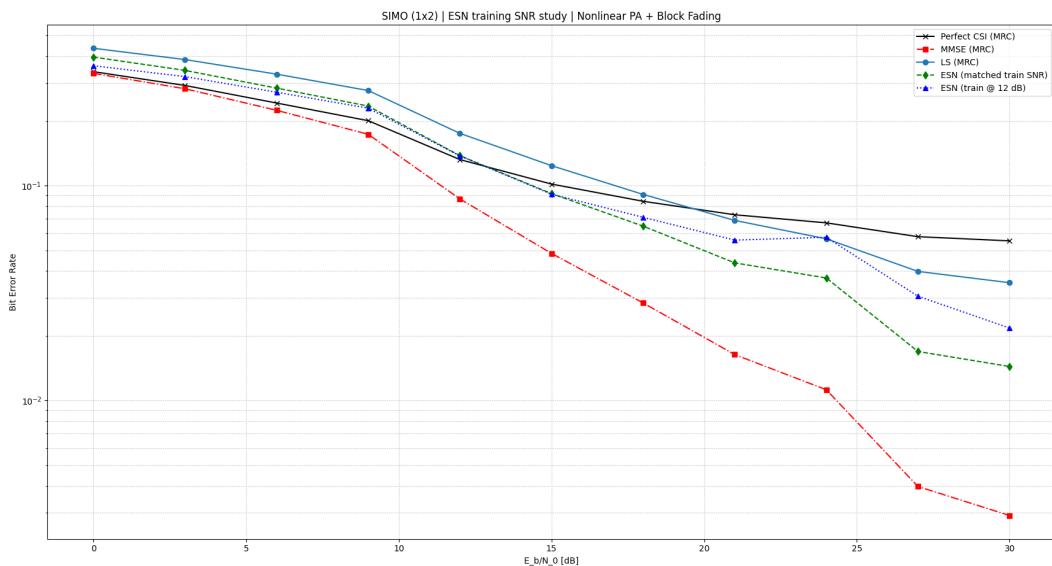


Figure 4.6: SIMO  $1 \times 2$ : uncoded BER vs.  $E_b/N_0$ . **All linear baselines use ZF detection**, differing only in the channel estimate: *Perfect-CE*, *LS-CE*, and *LMMSE-CE*. Learning curves are shown for *ESN (matched)* and *ESN (train @ 12 dB)*.

**Results** Moving from the SISO to the  $1 \times 2$  case brings the expected diversity gain: all BER curves shift noticeably to the left. The *matched* ESN closely tracks the LS-CE (ZF) baseline across the whole SNR range and stays reasonably near the LMMSE-CE (ZF)

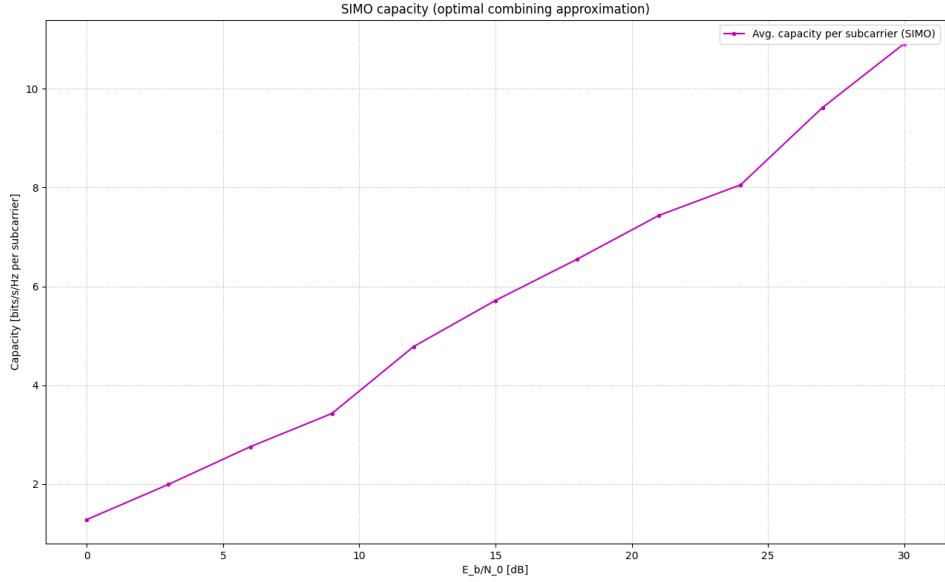


Figure 4.7: SIMO  $1 \times 2$ : average per-subcarrier capacity,  $C = \mathbb{E}_k \left[ \log_2 \left( 1 + \frac{\mathbf{h}_k^H \mathbf{h}_k}{N_0} \right) \right]$ , obtained from the same channel ensemble.

curve at low and mid SNR. When trained only once at 12 dB, the ESN shows some loss but remains stable rather than breaking down, staying within a moderate gap up to the mid-SNR region. The capacity plot confirms the expected steady growth with  $E_b/N_0$ , reflecting the combining gain from  $\|\mathbf{h}_k\|_2^2$ .

(i) With only one transmit stream, the ZF baselines set a clear reference point. The time-domain ESN essentially mirrors LS-CE (ZF) and stays surprisingly close to LMMSE-CE (ZF), despite not using explicit CSI. (ii) The leftward shift compared with SISO matches the capacity increase one would expect from the extra receive antenna, scaling with  $\|\mathbf{h}_k\|_2^2$ . (iii) At high SNR, LMMSE-CE (ZF) keeps its edge thanks to its noise-aware nature. That advantage, however, tends to diminish once coding is applied, as the later MIMO results will show.

#### 4.2.4 MIMO $2 \times 2$ Block-Fading (16-QAM)

We now move from receive diversity to true spatial multiplexing with two transmit and two receive antennas ( $N_t=2$ ,  $N_r=2$ ). The OFDM numerology and channel assumptions remain consistent with the earlier setups:  $N=512$  subcarriers, cyclic prefix  $N_{cp}=7$ , block fading with length  $L=8$  under an exponential PDP, Doppler shift  $f_D=100$  Hz, and a soft PA nonlinearity of +3 dB.

The learning receiver is an ESN that takes as input the stacked time-domain I/Q samples from both receive branches ( $2N_r$  inputs) and jointly predicts the two transmitted streams ( $2N_t$  outputs). We again compare a *matched-SNR* configuration with a *fixed-train @ 12 dB* setup.

Table 4.6: Stage—MIMO  $2 \times 2$  (short configuration).

Item	Setting
Antennas	$N_t=2, N_r=2$
OFDM / Mod.	$N=512, N_{\text{cp}}=7; 16\text{-QAM} (m=4)$
Channel	Block-fading, $L=8$ (exp. PDP), $f_D=100$ Hz
Noise / PA	$N_0=10^{-5}$ ; smooth clip +3 dB ( $p=1$ )
Training	Per-block pilot; ESN <i>matched</i> to test SNR & <i>fixed</i> @ 12 dB
ESN	Inputs $2N_r$ , outputs $2N_t$ , $n_{\text{res}}=150$ , $\rho=0.9$ ; delay search $0:\lceil L/2 \rceil+2$
Baselines	ZF (Perfect-ZF), ZF (LS-ZF), MMSE (TD MMSE $\rightarrow$ LS-ZF)
Metrics	Uncoded BER vs. $E_b/N_0$ ; capacity & usable rank from per-tone SVD

**Results** Compared with the SIMO case, the  $2 \times 2$  MIMO setup introduces true spatial multiplexing, which requires stronger separation of the transmitted streams.

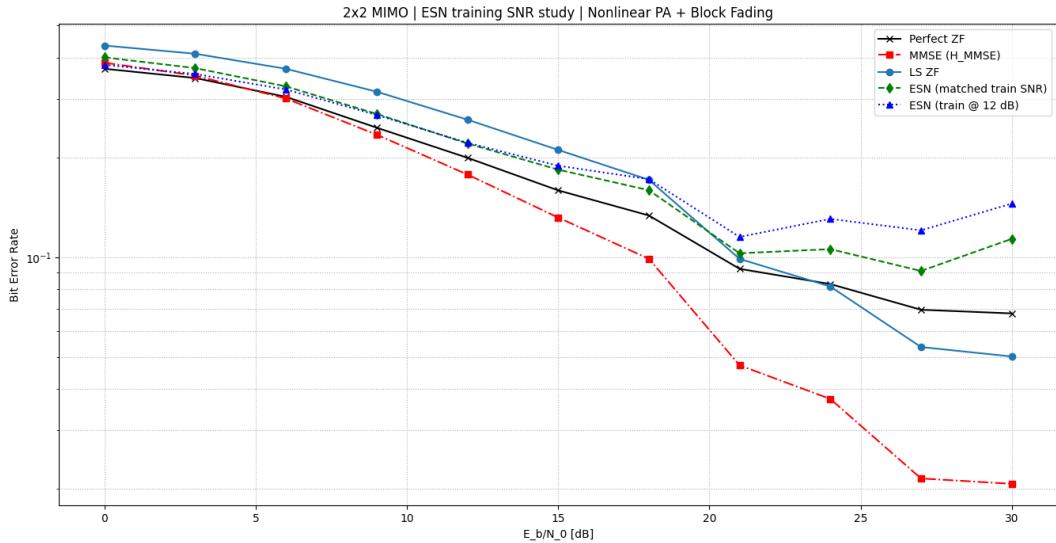


Figure 4.8:  $2 \times 2$  MIMO, 16-QAM, nonlinear PA: uncoded BER vs.  $E_b/N_0$  for Perfect ZF, LS-ZF, MMSE, and ESN (matched and fixed-train@12 dB).

- (i) Among the baselines, **MMSE** clearly performs best across the entire SNR range, showing the benefit of noise-aware estimation. Perfect-ZF approaches MMSE as noise vanishes at higher SNR, while LS-ZF remains consistently weaker due to its less accurate channel estimates.
- (ii) The *matched* ESN tracks LS-ZF closely at low and mid SNR, but diverges in the high-SNR region, reflecting the additional burden of handling inter-stream interference without explicit CSI. The *fixed* 12 dB model suffers from mismatch, though the degradation is gradual and it remains functional.
- (iii) The capacity curve increases monotonically with SNR and reaches beyond 16 bits/s/Hz per subcarrier at 30 dB. The fraction of subcarriers with usable rank  $\geq 2$  stays nearly one

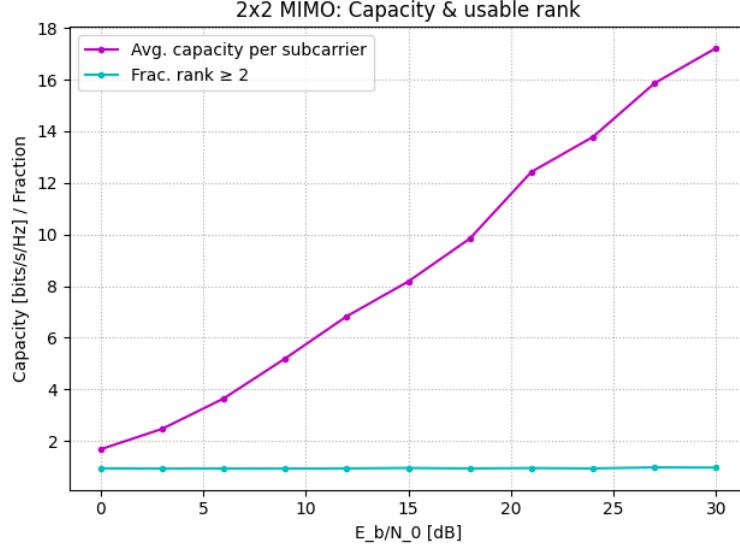


Figure 4.9:  $2 \times 2$  MIMO capacity (average per subcarrier) and fraction of usable rank  $\geq 2$  obtained from the same  $\mathbf{H}_k$  ensemble.

across the full range, confirming that the  $2 \times 2$  channel is effectively full-rank under the CDL-B fading model.

#### 4.2.5 MIMO $4 \times 8$ (16-QAM) with LDPC: Training-SNR

We now extend the setup to a large-scale configuration with four transmit and eight receive antennas. This stage stresses the reservoir under significant inter-stream interference and examines (i) the robustness of training-SNR alignment, and (ii) the effect of LDPC decoding in compressing symbol-level gaps into smaller post-decoder differences.

Table 4.7: System Model II — Stage 4 (MIMO  $4 \times 8$ ) short configuration.

Item	Setting
Scenario	$4 \times 8$ MIMO-OFDM, block-fading; soft-PA (+3 dB, smooth clip).
OFDM & channel	$N = 512$ subcarriers, $N_{\text{cp}} = 7$ ; $L = 8$ taps (exp. PDP), $f_D = 100$ Hz.
Modulation & pilots	16-QAM ( $m = 4$ ); per-coherence-block pilots. FDM pilot allocation: tone $k$ assigned to TX $t$ if $k \equiv t \pmod{4}$ .
Baselines	Perfect-ZF; LS-ZF; TD-MMSE $\rightarrow$ MMSE detector.
ESN demapper	$n_{\text{res}} = 60$ ; spectral radius 0.9; stacked TD I/Q across 8 receivers as input, TD I/Q for 4 streams as output; delay search $0 - \lceil L/2 \rceil + 2$ . Two regimes: <i>matched</i> train-SNR and <i>fixed</i> @ 12 dB.
Coding	( $d_v=4, d_c=8$ ) regular LDPC; one codeword per OFDM symbol/stream ( $n = mN$ bits, rate $\approx 0.5$ ); max-log LLR with clipping; 80 iters (FAST).
Metrics	Pre-/post-LDPC BER vs. $E_b/N_0$ ; SVD-based channel rank and capacity (logged).

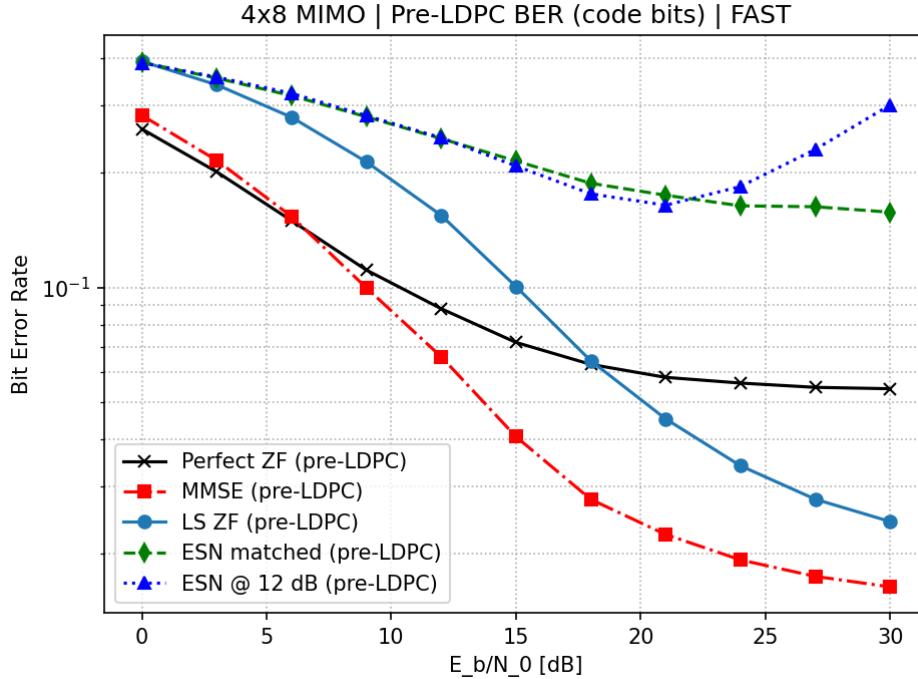


Figure 4.10: MIMO  $4 \times 8$ : uncoded (pre-LDPC) BER vs.  $E_b/N_0$  for Perfect ZF, LS-ZF, MMSE, and ESN (matched and fixed @12 dB).

**Results.** The  $4 \times 8$  configuration introduces stronger inter-stream interference compared to the lower-dimensional cases, and the BER curves reflect this increased complexity. In the uncoded case (Fig. 4.10), MMSE detection achieves the lowest error rates

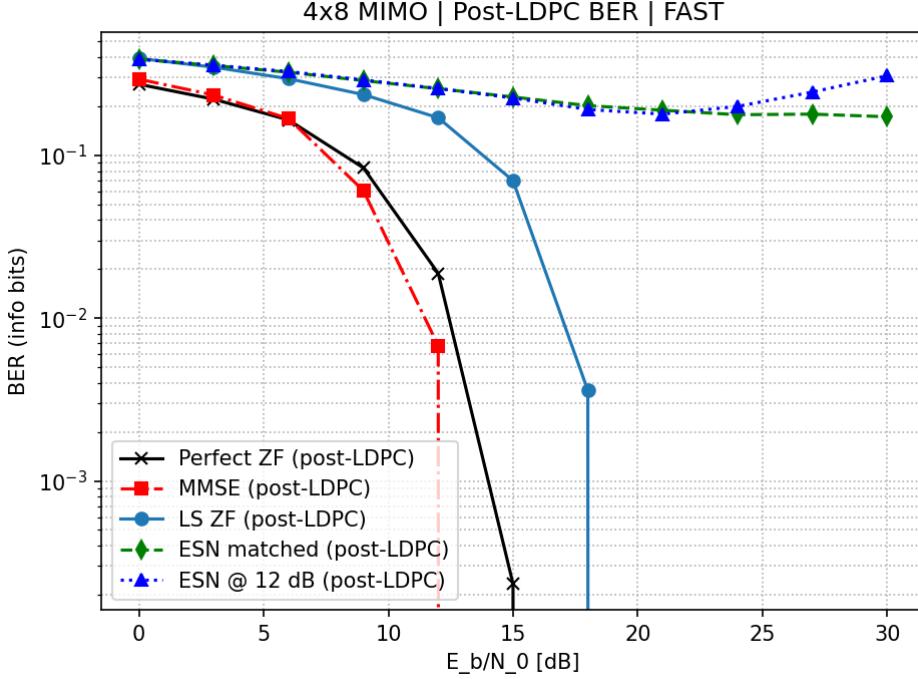


Figure 4.11: MIMO  $4 \times 8$ : coded (post-LDPC) BER vs.  $E_b/N_0$ . LDPC decoding compresses performance gaps among linear baselines but ESN remains non-competitive.

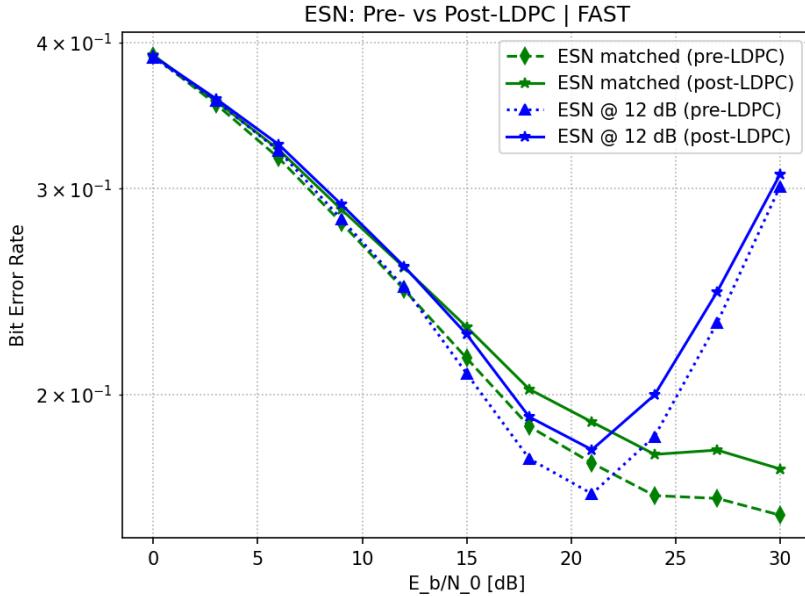


Figure 4.12: MIMO  $4 \times 8$ : ESN-only comparison (pre- vs. post-LDPC) for matched and fixed 12 dB training. LDPC reduces BER somewhat but cannot close the gap to baselines.

across the SNR range, with Perfect-ZF following closely and LS-ZF slightly degraded. The ESN trained in a matched-SNR setting tracks LS-ZF reasonably well at low to mid SNRs, but its performance saturates at higher SNR where an error floor emerges. The fixed-training ESN at 12 dB performs consistently worse, showing a clear mismatch

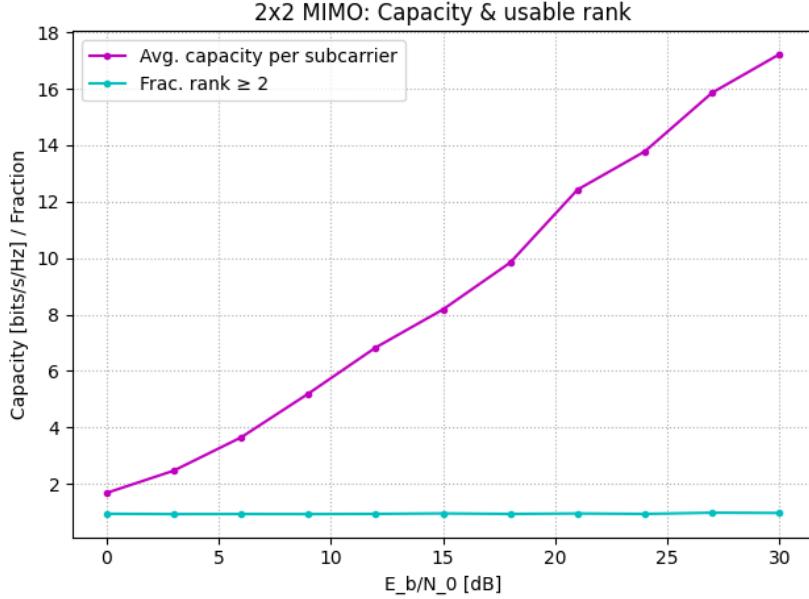


Figure 4.13:  $4 \times 8$  MIMO capacity (average per subcarrier) and fraction of usable rank  $\geq 4$  obtained from per-subcarrier SVD.

penalty across the range.

After LDPC decoding (Fig. 4.11), the gap between the baselines and the ESN widens further. MMSE benefits strongly from coding, reaching near error-free operation from around 12–15 dB upwards, while Perfect-ZF converges similarly. LS-ZF also improves significantly, albeit requiring higher SNR to reach low BER values. The ESN, however, remains limited by its pre-decoder error floor, and although coding reduces BER slightly at intermediate SNRs, both the matched and fixed models remain far above the baselines. Figure 4.12 highlights this behavior, showing that LDPC decoding provides only partial relief for ESN errors and that the overall curve shape remains dominated by systematic symbol-level inaccuracies.

The channel analysis in Fig. 4.13 shows that the average per-subcarrier capacity increases steadily with SNR, exceeding 40 bits/s/Hz at 30 dB, reflecting the richness of the  $4 \times 8$  CDL channel. At the same time, the fraction of subcarriers with usable rank  $\geq 4$  stays very low, only reaching a small fraction at higher SNR. This indicates that although the channel supports high theoretical capacity, in practice many tones are effectively constrained to fewer active spatial streams. These conditions are well handled by structured linear detectors such as MMSE but remain challenging for the ESN, whose error floor is consistent with the difficulty of separating multiple interfering streams in this high-rank setting.

#### 4.2.6 MIMO $4 \times 8$ with CDL (16-QAM) and LDPC: Training-SNR

**Results (concise).** The results in Fig. 4.14 confirm that under CDL-B fading, the linear MMSE baseline remains the most effective. Its uncoded curve already dominates, and when combined with LDPC decoding using calibrated LLRs, it exhibits the expected sharp waterfall, pushing BER well below  $10^{-3}$  around 15 dB.

Table 4.8: Stage—MIMO  $4 \times 8$  over CDL-B (short configuration).

Item	Setting
Antennas	$N_t=4, N_r=8$
OFDM / Mod.	$N=512, N_{cp}=15; 16\text{-QAM } (m=4)$
Channel	3GPP CDL-B (TDL), block fading; $W=2.048$ MHz; $L=16$ taps, DS $\approx 300$ ns
Noise / PA	Two-sided $N_0=10^{-5}$ ; smooth clip +3 dB (shape $p=1$ )
Training	Per-coherence-block pilot; ESN trained at test SNR (matched)
ESN demapper	Inputs $2N_r$ , outputs $2N_t$ ; $n_{\text{res}} \sim 300$ , spectral radius $\rho=0.9$ ; I/Q delay search $0:[L/2]+2$
Baselines	Linear MMSE (per-tone) with TD-MMSE channel es- timation
Coding	Systematic LDPC (rate $\approx 0.5$ ); bit-LLRs via max-log + scalar temperature calibration; $ \text{LLR} $ clipping
Metrics	Uncoded & post-LDPC BER vs. $E_b/N_0$ ; avg. capac- ity & fraction of usable rank $\geq 4$ (per-tone Singular Value Decomposition (SVD))

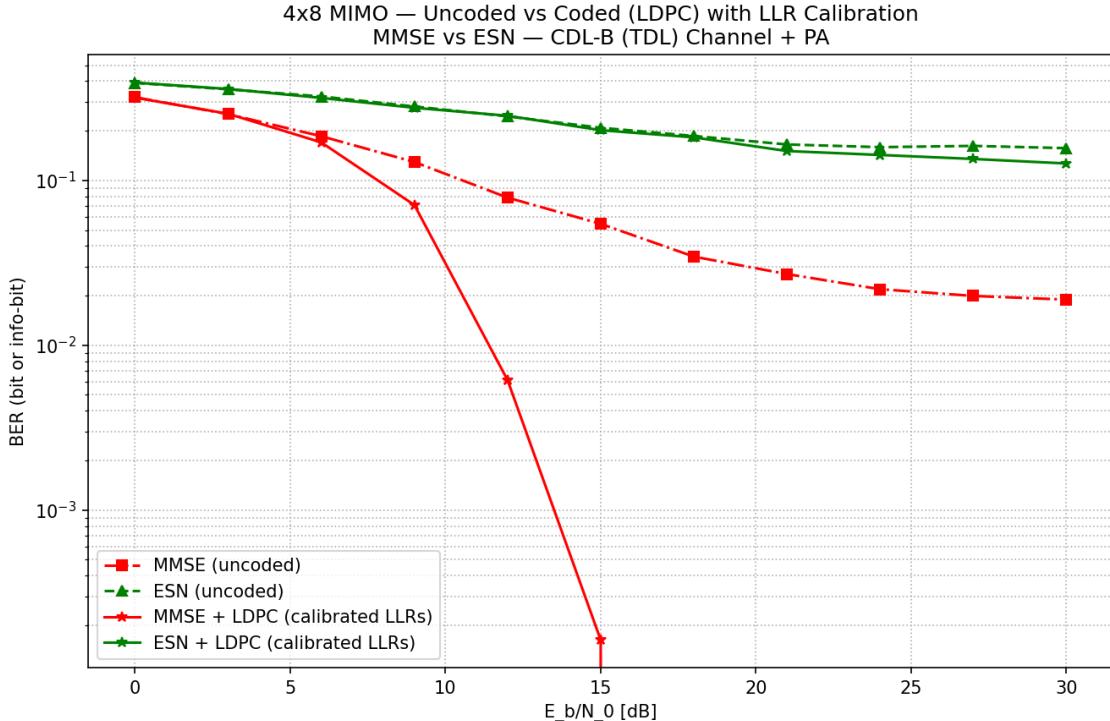


Figure 4.14:  $4 \times 8$  MIMO over CDL-B: uncoded vs. coded (LDPC) BER with LLR cali-  
bration; comparison of ESN demapper and linear MMSE.

The ESN demapper, while competitive with MMSE at very low SNR in the uncoded case, falls behind quickly as SNR increases. After coding, the ESN improves over its uncoded performance but fails to deliver substantial coding gain: its post-LDPC curve saturates around  $10^{-1}$ , highlighting that the decoder cannot fully exploit its soft

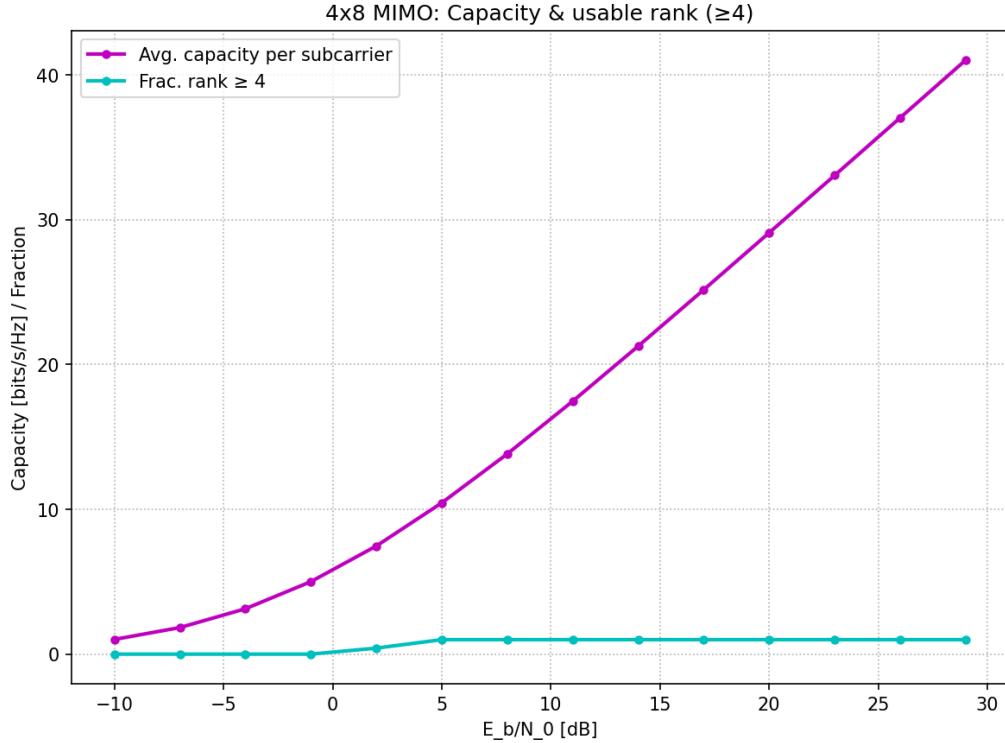


Figure 4.15:  $4 \times 8$  MIMO over CDL-B: average per-subcarrier capacity and fraction of subcarriers with usable rank  $\geq 4$ .

outputs.

Figure 4.15 shows that this gap is not due to spatial limitations. The channel capacity rises steeply with SNR, reaching over 40 bits/s/Hz at 30 dB, and nearly all subcarriers become rank-4 by mid-SNR. Thus, the bottleneck lies in the demapper's ability to provide reliable soft information rather than in channel richness.

*Takeaways.* (1) The CDL-B scenario reinforces the strength of MMSE: with well-calibrated LLRs, it benefits fully from LDPC coding and achieves steep waterfalls. (2) The ESN demonstrates limited coding gain. While it reduces errors modestly after LDPC, its poor calibration prevents the decoder from achieving large improvements. (3) These results suggest that progress requires moving beyond hard-decision training: either a soft-output ESN, refined LLR calibration, or hybrid ESN+linear schemes may be needed to unlock the full potential in CDL channels.

#### 4.2.7 MIMO $2 \times 2$ with ML-Based Detectors: Model Comparison

Table 4.9: Stage—MIMO  $2 \times 2$  (ML model comparison; short configuration).

Item	Setting
Antennas	$N_t=2, N_r=2$
OFDM / Mod.	$N=512, N_{\text{cp}}=7$ ; 16-QAM ( $m=4$ )
Channel	Block-fading, $L=8$ (exp. Power Delay Profile (PDP)), $f_D=100$ Hz
Noise / PA	$N_0=10^{-5}$ ; smooth clip +3 dB ( $p=1$ )
Training	Per-block pilots; each model learns TD $x_{\text{cp}}$ from stacked TD $y_{\text{cp}}$
ESN	Inputs $2N_r$ , outputs $2N_t$ ; $n_{\text{res}}=100, \rho=0.9$ ; fixed delay = 3
CNN	1D conv: $4 \rightarrow 32 \rightarrow 64 \rightarrow 4$ (kernel 5, ReLU), 50 epochs
RNN	LSTM( $4 \rightarrow 100$ ) + FC $\rightarrow 4$ , 50 epochs
FFNN	Sliding window $w=8$ (input $4w$ ), FC $4w \rightarrow 100 \rightarrow 4$ , 50 epochs
ELM	Sliding window $w=8$ ; tanh hidden 100; closed-form readout
Baselines	ZF (Perfect-CSI), ZF (LS- $\hat{\mathbf{H}}$ ), MMSE (TD-MMSE $\rightarrow \hat{\mathbf{H}}$ )
Metric	Uncoded BER vs $E_b/N_0$

**Results and discussion.** Figure 4.16 overlays uncoded BER for the ESN and four Neural Network (NN)-based baselines (CNN, RNN, FFNN, and ELM) against the classical ZF and MMSE references.

Across the SNR sweep, MMSE remains the strongest linear baseline. The ESN consistently improves upon LS-ZF and stays within roughly 1~3 dB of MMSE at mid-high SNR, suggesting that reservoir dynamics capture useful temporal dependencies missing from purely linear methods.

The CNN and RNN models, trained with the same per-block pilots, manage to close part of the gap to MMSE but remain slightly behind the ESN. This may be attributed to their fixed output delay and lack of explicit input normalisation. The windowed FFNN and ELM underperform at low SNR and only improve gradually as SNR increases, never quite reaching the accuracy of ESN or MMSE.

Overall, the ESN offers a favourable trade-off between accuracy and complexity for this block-fading  $2 \times 2$  MIMO setting. That said, the MMSE detector continues to serve as a tough benchmark, underscoring that carefully designed linear baselines are still remarkably competitive.

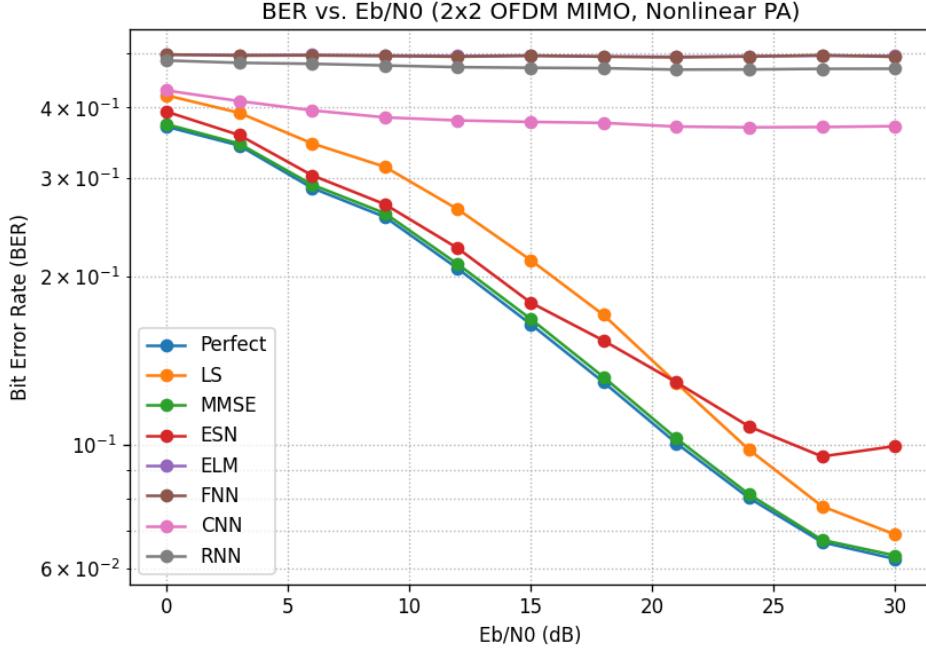


Figure 4.16: MIMO  $2 \times 2$  (block fading): uncoded BER vs.  $E_b / N_0$  for ESN, CNN, RNN, FFNN, ELM, and classical baselines.

### 4.3 ESN Hyperparameter Selection (System Model II)

To identify a robust ESN configuration, we performed controlled sweeps in which *one* hyperparameter was varied at a time while the others were kept fixed. Each configuration was then evaluated over  $E_b / N_0 \in [0, 30]$  dB using the same  $2 \times 2$  block-fading MIMO-OFDM pipeline (nonlinear PA,  $L=8$  taps, 16-QAM). The sweep driver (`pyesn_parameter_sweep.py`) explored:

- reservoir size  $\{50, 100, 200, 400\}$ ,
- spectral radius  $\{0.7, 0.9, 1.1\}$ ,
- input scaling  $\{10^{-3}, 5 \times 10^{-3}, 2 \times 10^{-2}\}$ ,
- sparsity  $\{0.1, 0.3, 0.5\}$ ,
- teacher scaling  $\{5 \times 10^{-7}, 10^{-6}, 5 \times 10^{-6}\}$ .

For each setting, uncoded BER curves were logged across the SNR sweep. The script mirrors the training and inference flow used in the main experiments: pilot-based training per coherence block, TD input/TD output, and FFT to frequency domain for hard decisions.

#### Key observations.

- **Reservoir size.**  $N_{\text{res}}=100$  consistently gave the lowest BER at high SNR and matched or surpassed  $N_{\text{res}}=200$  at mid SNR. Smaller reservoirs (50) underfit, while very large ones (400) tended to overfit and even amplify noise.

- **Spectral radius.**  $\rho=0.9$  proved best across the sweep. A smaller value (0.7) limited memory depth, while 1.1 pushed the reservoir toward instability, showing signs of violating the Echo State Property (ESP).
- **Input scaling.** Larger input drive improved performance.  $\text{inpscale}=0.02$  clearly boosted mid–high SNR accuracy, whereas very small values (e.g.,  $10^{-3}$ ) left the reservoir under-stimulated.
- **Sparsity.** Moderate sparsity worked well.  $\text{sparsity}=0.3$  offered the best across-SNR compromise; both very sparse and very dense connectivity led to small but consistent degradations.
- **Teacher scaling.** A small ridge term stabilised the readout. Values around  $5\times 10^{-7}$ – $10^{-6}$  were most effective; larger scaling ( $5\times 10^{-6}$ ) slightly flattened high-SNR performance.

**Chosen default.** Balancing accuracy and computational cost, we adopt the following ESN setup for subsequent experiments:

$$N_{\text{res}} = 100, \quad \rho = 0.9, \quad \text{inpscale} = 0.02, \quad \text{sparsity} = 0.3, \quad \text{teacher} = 5\times 10^{-7}.$$

This configuration yields strong BER in the mid–high SNR region while avoiding excessive state size or unstable training sensitivity.

Table 4.10: Final ESN hyperparameters (selected from parameter sweeps).

Hyperparameter	Value	Rationale
Reservoir size $N_{\text{res}}$	100/150	Best BER at high SNR; avoids overfitting and overhead
Spectral radius $\rho$	0.9	Near the ESP edge; stable yet expressive
Input scaling	0.02	Ensures sufficient drive; very small values under-excite states
Sparsity	0.3	Balanced accuracy–complexity trade-off
Teacher scaling	$5\times 10^{-7}$	Stabilises readout; avoids high-SNR floor

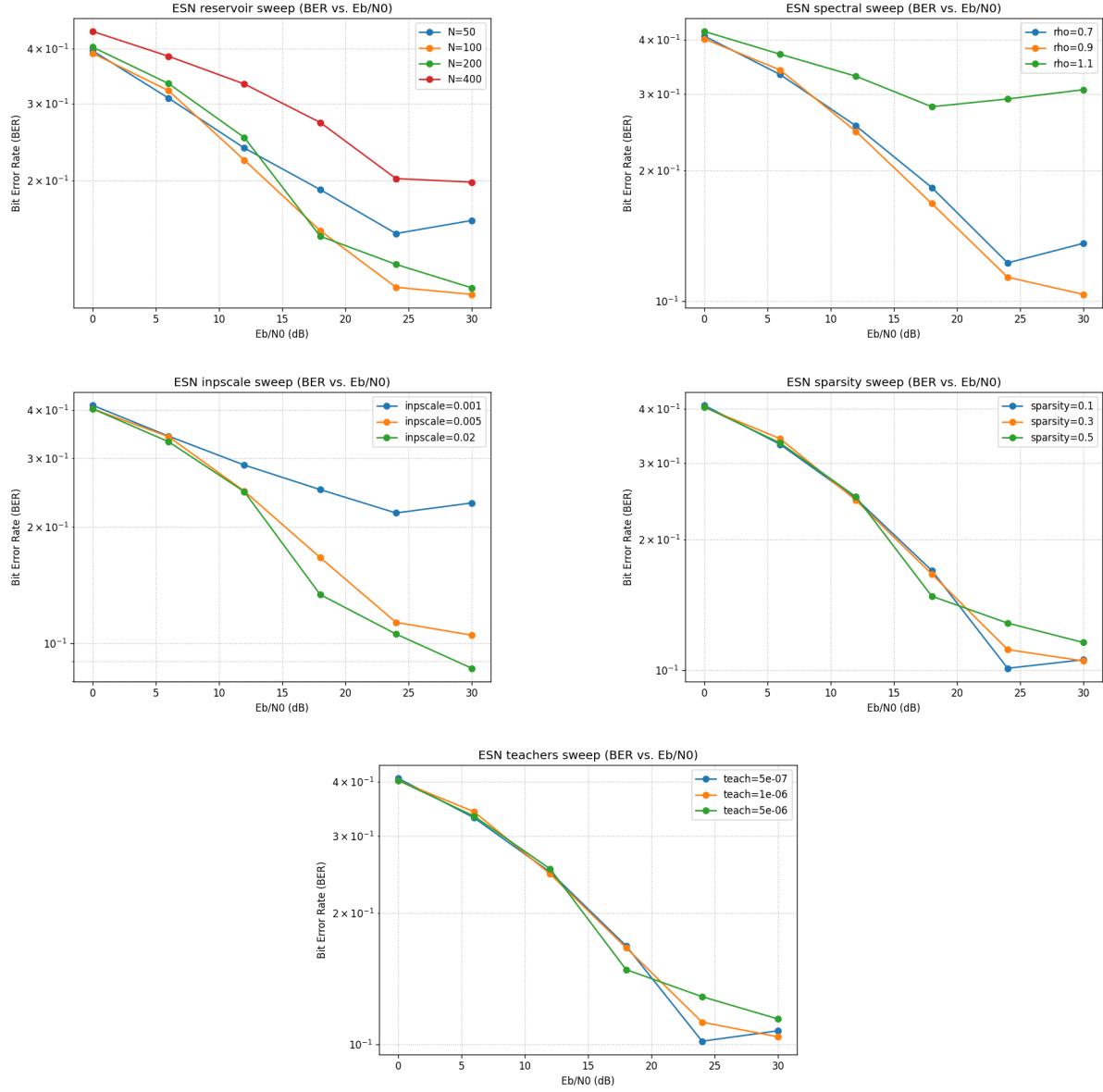


Figure 4.17: One-at-a-time ESN sweeps: reservoir size, spectral radius  $\rho$ , input scaling, sparsity, and teacher scaling (BER vs.  $E_b / N_0$ ).

# **Conclusion and Future Directions**

**4.4 Summary of key findings**

**4.5 Contributions to the field**

**4.6 Study limitations**

**4.7 Recommendations for future research**

**4.8 Final Thoughts**

# Bibliography

- [1] A. Taneja, N. Alqahtani, and A. Alqahtani, "Interference aware resource control for 6g-enabled expanded iot networks," *Sensors*, vol. 23, no. 12, p. 5649, 2023. [Online]. Available: <https://doi.org/10.3390/s23125649>
- [2] K. Peppas, H. Nistazakis, and G. Tombras, "An overview of the physical insight and the various performance metrics of fading channels in wireless communication systems," *Advanced trends in wireless communications*, vol. 13, pp. 1–22, 2011.
- [3] B. Baumgärtner, Tiger66, and T. Frei, "Mimo simo miso siso explanation without confusion," [https://commons.wikimedia.org/wiki/File:MIMO\\_SIMO\\_MISO\\_SISO\\_explanation\\_without\\_confusion.svg](https://commons.wikimedia.org/wiki/File:MIMO_SIMO_MISO_SISO_explanation_without_confusion.svg), 2018, licensed under CC BY-SA 3.0.
- [4] Z. Zhou, L. Liu, V. Chandrasekhar, J. Zhang, and Y. Yi, "Deep reservoir computing meets 5g mimo-ofdm systems in symbol detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 1266–1273.
- [5] S. Mosleh, L. Liu, C. Sahin, Y. R. Zheng, and Y. Yi, "Brain-inspired wireless communications: Where reservoir computing meets mimo-ofdm," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4694–4708, 2018.
- [6] S. Rajendran, T. J. O'Shea, T. C. Clancy, and R. W. McGwier, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, 2018.
- [7] J. Hoydis, S. ten Brink, and M. Debbah, "Channel estimation techniques for pilot contamination mitigation in massive mimo systems," *IEEE Journal on Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 837–850, 2012.
- [8] H. Jaeger, "The “echo state” approach to analysing and training recurrent neural networks," *GMD Report*, vol. 148, 2001.
- [9] M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Computer Science Review*, vol. 3, no. 3, pp. 127–149, 2009.
- [10] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

- [11] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "Training recurrent neural networks," *Advances in Neural Information Processing Systems*, vol. 26, 2013.
- [12] D. Verstraeten, B. Schrauwen, M. D'Haene, and D. Stroobandt, "An experimental unification of reservoir computing methods," *Neural Networks*, vol. 20, no. 3, pp. 391–403, 2007.
- [13] Z. Zhou, L. Liu, S. Jere, J. Zhang, and Y. Yi, "Deep reservoir computing meets 5g mimo-ofdm systems in symbol detection," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 1, pp. 1266–1273, 2020.
- [14] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *Science*, vol. 304, no. 5667, pp. 78–80, 2004.
- [15] M. D. Skowronski and J. G. Harris, "Automatic speech recognition using a predictive echo state network classifier," *Neural Networks*, vol. 20, no. 3, pp. 414–423, 2006.
- [16] H. Jaeger, "Short term memory in echo state networks," in *GMD-Forschungszentrum Informationstechnik*, vol. 5, 2002, pp. 61–73.
- [17] D. Bush and S. Anderson, "Echo state networks for self-localization and mapping," in *IEEE Congress on Evolutionary Computation*. IEEE, 2005.
- [18] S. Tong, T. Peng, and L. Wang, "Echo state networks for language modeling," in *International Conference on Neural Information Processing*. Springer, 2007, pp. 975–984.
- [19] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5g cellular: It will work!" *IEEE access*, vol. 1, pp. 335–349, 2013.
- [20] C.-X. Wang, F. Haider, X. Gao, X.-H. You, Y. Yang, D. Yuan, H. M. Aggoune, H. Haas, S. Fletcher, and E. Hepsaydir, "Cellular architecture and key technologies for 5g wireless communication networks," *IEEE communications magazine*, vol. 52, no. 2, pp. 122–130, 2014.
- [21] H. Jaeger, "The "echo state" approach to analysing and training recurrent neural networks," *GMD Report*, vol. 148, 2001.
- [22] S. Mosleh, L. Liu, C. Sahin, Y. R. Zheng, and Y. Yi, "Where reservoir computing meets mimo-ofdm," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4694–4707, 2018.
- [23] M. Bauduin, "Equalization of non-linear satellite communication channels using echo state networks," Ph.D. dissertation, Université Libre de Bruxelles, 2016.
- [24] B. W. Sun, "Estimation of snr in ka-band satellite acm communication system," Master's thesis, GlobeThesis Open Access, 2019.
- [25] M. Giordani, M. Polese, and et al., "Toward 6g networks: Use cases and technologies," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 55–61, 2020.

- [26] P. Popovski *et al.*, "Wireless access beyond 5g: A vision of 6g," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 1–7, 2019.
- [27] H.-P. Yin, C. Bai, and H.-P. Ren, "Echo state network based symbol detection in chaotic baseband wireless communication," *arXiv preprint arXiv:2103.08159*, 2021.
- [28] P. Lohan, B. Kantarci, M. Amine Ferrag, N. Tihanyi, and Y. Shi, "From 5g to 6g networks: A survey on ai-based jamming and interference detection and mitigation," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 3920–3974, 2024.
- [29] C. Mingzhe, W. SAAD, and Y. Changchuan, "Echo state learning for wireless virtual reality resource allocation in uav-enabled lte-u networks [c]," in *2018 IEEE International Conference on Communications, Kansas City, USA*, 2018, pp. 1–6.
- [30] S. K. Moorthy, N. Mastronarde, S. Pudlewski, E. S. Bentley, and Z. Guan, "Swarm uav networking with collaborative beamforming and automated esn learning in the presence of unknown blockages," *Computer Networks*, vol. 231, p. 109804, 2023.
- [31] T. J. O'Shea and J. Hoydis, "An overview of deep learning for communications systems," *arXiv preprint arXiv:1701.08856*, 2017.
- [32] R. Giuliano and E. Innocenti, "Machine learning techniques for non-terrestrial networks," *Electronics*, vol. 12, no. 3, p. 652, 2023.
- [33] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in ofdm systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, 2018.
- [34] N. Farsad, M. Rao, and A. Goldsmith, "Deep learning for joint channel estimation and signal detection in ofdm systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 10, pp. 6262–6276, 2018.
- [35] Z. Wu *et al.*, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, 2020.
- [36] Q. Huang, X. Xie, and M. Cheriet, "Reinforcement learning for resource allocation in wireless networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 250, pp. 1–12, 2020.
- [37] M. Giordani, M. Polese, A. Roy, S. Basagni, and M. Zorzi, "Toward 6g networks: Use cases and technologies," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 55–61, 2020.
- [38] M. Steck, P. Zimmermann, and H. Barzegar, "Echo state network equalization for leo satellite ofdm links," in *Proc. IEEE ICC*, 2023, pp. 1–6.
- [39] Z. Zhang *et al.*, "Revealing untapped dsp optimization potentials for fpga-based systolic matrix engines," *arXiv preprint arXiv:2409.03508*, 2024. [Online]. Available: <https://arxiv.org/abs/2409.03508>
- [40] Z. Zhang, "Photonic reservoir computing for equalizing phase-modulated signals," Master's thesis, Ghent University, 2021.

# **Appendix**

# **Appendix 1 - Code Embed**

**Python Code**

**Acknowledgment**

**List of Abbreviations and Acronyms**