Low-complexity machine learning-based equalisation in optical communication systems

Karina Nurlybayeva
Doctor of Philosophy

Aston Institute of Photonic Technologies
Aston University
June 2023

June, 2023

Aston University

Low-complexity machine learning-based equalisation in optical communication systems

Karina Nurlybayeva

Doctor of Philosophy, June 2023

Kerr nonlinearity has a significant degrading effect on the performance of high-speed optical transmission systems. Hence, to achieve reliable and high-quality optical communication, compensating for fibre nonlinearity is crucial. However, many existing techniques like Digital back-propagation (DBP), Inverse Volterra series transfer functions (IVSTF), Nonlinear Fourier Transform (NFT), and Optical phase conjugation (OPC) have drawbacks, such as high computational complexity, marginal performance benefits, difficulty in reconfiguring.

In recent years, machine learning (ML) has gained popularity as a promising solution for the compensation of nonlinear fibre effects. This is primarily due to the fact that Machine Learning (ML) techniques are universal approximations, allowing them to reverse the channel propagation function and effectively mitigate fibre impairments. Additionally, data science approaches like ML in optical communication applications excel due to the high data availability. However, a major challenge faced by most ML-based equalisation implementations is the high computational complexity, which imposes significant demands on device speed and energy consumption during equalisation operations. This becomes more prominent during training, requiring much energy and training data.

This thesis explores low-complexity machine learning implementations, including Support Vector Machine (SVM), Support Vector Regression (SVR), and low-complexity Neural Networks (NN), particularly complex-valued multilayer perceptron. The impact of these approaches is evaluated through bit error rate (BER) performance and resulting computational complexity.

Moreover, the thesis investigates different strategies for simplifying the resulting equalisers or the training process while maintaining adequate performance. Techniques such as weight pruning, integration with optical solutions (like optical phase conjugation and dispersion management), and crowd equalisation based on committee learning are examined.

The findings provide valuable insights into their performance, computational complexity, and potential enhancements, thereby contributing to the development of efficient and effective equalisation techniques in optical communication.

Keywords and phrases: Optical communications, Nonlinearity Mitigation, Machine learning, Neural Networks

Acknowledgements

I want to express my deepest gratitude and appreciation to my wonderful supervisors, Prof Sergei K. Turitsyn and Dr Elena Turitsyna, for their invaluable guidance, support, and supervision throughout the completion of this thesis. This research project would not have been possible without their dedication and expertise.

Furthermore, I would like to express my deepest gratitude to my mentor, Dr Morteza Kamalian-Kopae, for his ongoing availability and willingness to engage in meaningful discussions. His open-door policy and willingness to respond to my concerns and questions at any time were invaluable. I would also like to thank him for his amazing ideas and discussions. These discussions helped me to develop some ideas presented in this thesis. I will always cherish the knowledge and skills I have gained through his mentorship, which I am confident will continue to impact my future endeavours.

I would also like to extend my gratitude to the entire AiPT community and my fellow colleagues who provided support and contributed to the development of this research. Their insights and collaboration have been instrumental in shaping my understanding and broadening the scope of this work. Especially, I would like to thank my collaborator Diego Argüello Ron for his help in developing the pruning and multi-symbol output codes and Prof Andrew Ellis' team for their contribution to the results for the mid-link Optical Phase conjugation. I also want to acknowledge support from the EPSRC Core Equipment Fund, Grant EP/V036106/1, for providing the Aston EPS Machine Learning Server and thank Dr Thomas Carr for his invaluable help in navigating this server and for offering valuable advice on the intricacies of machine learning.

Also, I sincerely thank Mohammed Patel, Swaroopa Mucheli-Sudhakar, Dr Pratim Hazarika, Dr Aleksandr Donodin and Dr Florent Bessin for teaching me the basics of conducting experiments in the labs and for their insightful and productive discussions about optical communications, which have significantly enriched my academic journey. I also thank my fellow PhD peers, Dr Paulami Ray, Dr Marie Zandi, Dr Nasir Bello, Dr Namita Sahoo, and Dr Anastasiia Vasylchenkova, for their mental support during the challenging period of the Covid pandemic. Also, thanks to Aisha Bibi for being the best office mate I

could ever dream of. Throughout this journey, she has become like a second family to me.

I am also very grateful to my wonderful friends, Annelise Garrison and Sofiia Koviarova, who were always there despite all the challenges.

Lastly, I sincerely thank my family and friends for their presence throughout this journey. Their belief in me and their love sustained me during challenging times. My heartfelt gratitude goes to my mother, Aigul Sainova, whose unwavering support and encouragement made this accomplishment possible. I am forever grateful for her indispensable role in my academic pursuits.

To my mom

Contents

List of Figures

List of Tables

List of Abbreviations

ANN Artificial Neural Network

ASE Amplified Spontaneous Emissions

BER Bit Error Rate

CC Computational Complexity

CD Chromatic Dispersion

CVNNs Complex-valued Neural Networks

CW Continuous-wave

DBP Digital back-propagation

DCF Dispersion compensating fibre

DCM Dispersion Compensating module

DM Dispersion Managed

DSP Digital Signal Processing

DUM Dispersion unmanaged

EDFA Erbium-doped fibre amplifier

EVM Error vector magnitude

FWM Four-wave mixing

HD-FEC Hard-decision Forward error correction

ISI Inter-symbol interference

IVSTF Inverse Volterra series transfer functions

LO Local oscillator

ML Machine Learning

MLP Multilayer Perceptron

MSE Mean-Squared error

NFT Nonlinear Fourier Transform

NLE Nonlinear equaliser

NLSE Nonlinear Schrodinger equation

NN Neural Networks

OPC Optical phase conjugation

PCTW Phase-conjugated twin-waves

PDM Polarisation division multiplexing

PMD Polarisation-mode dispersion

QAM Quadrature amplitude modulation

QoT Quality of transmission

ReLU Rectified Linear Unit

RRC Rsoot-raised cosine

SD-FEC Soft-decision Forward error correction

SMF Single-mode fibre

SNR Signal-to-Noise Ratio

SPM Self-Phase Modulation

SSMF Standard single-mode fibre

SVM Support Vector Machine

SVR Support Vector Regression

WDM Wavelength-division multiplexing

XPM Cross-Phase Modulation

Chapter 1

Introduction

Optical communication has become an essential part of modern society's technological progress[2]. It is affordable, capable of handling large amounts of data, and has numerous other alluring qualities. Recent years have seen advancements in optical system technology, including the development of lasers, amplifiers, fibres, coherence detection, digital signal processing, etc. Nevertheless, despite all the advancements made, there is a growing need for improved optical communication [3]. The industry has historically relied on hardware-based advancements, but over the past ten years, the demand for alternative software-based solutions has grown as the complexity of optical communication systems has increased.

Optical communication faces certain issues, such as the limitation of the maximum amount of information and transmission distance and degrades of signal-to-noise-ratio, which are introduced by fibre nonlinearity, caused by the Kerr effect [4]. The Kerr effect is a nonlinear phenomenon that causes the distortion of the propagated optical signal and is proportional to its power, leading to the data transmission decelerating [5]. This problem, along with the increased complexity of the system, results in several challenges that must be solved to maintain successful optical communication (reliability and quality of the system) without compromising the speed and data capacity[6].

Machine learning (ML) is a branch of Artificial Intelligence which can be described as a mathematical tool for decision-making by inferring statistical characteristics of the information being monitored. In recent years, machine learning has gained attraction as a promising technique for mitigating fibre nonlinearities in optical communication systems [5; 7]. The use of neural networks (NN), in particular, allows for the enhancement of existing fibre-optic systems without any prior knowledge of their characteristics. Despite its outstanding performance, such equalisers' computational cost is high, making real-world implementation impossible [8]. Additionally, the training process for such an equaliser requires a large amount of data and computational resources.

In this thesis, two ways to address these issues are explored: designing low-complexity equalisers and simplifying the training process, and making it more efficient, which would

result in a lower computational load and required electricity. We started by investigating intrinsically low-complexity solutions, like Support Vector Machines and Support Vector Regression. Then NN-based equalisation using Complex-valued Multilayer perceptron was investigated. Different ways to reduce the resulting complexity of the NNs-based equaliser were explored, such as the application of pruning, integrating optical solutions, and considering its photonic implementation.

For a more efficient learning stage, several techniques were suggested that either aim to achieve faster training, training with fewer data, or better utilisation of the available data.

1.1 Overview of the Thesis

This thesis is organised as follows:

Chapter 2 provides a brief explanation of challenges in optical communication and the need for fibre nonlinearity compensation, as well as discusses the advantages and disadvantages of the existing techniques. Then a brief background of machine learning is provided, followed by a discussion of the role of machine learning in optical communication and, in particular, nonlinearity equalisation.

Chapter 3 provides a brief overview of the Support Vector Machine and Support Vector Regression, followed by a demonstration of their implementation for nonlinearity mitigation and an analysis of the system's performance in terms of resulting BER performance and computational complexity.

In Chapter 4, the Neural Networks-based equalisation is suggested using Complex-Valued Multilayer perception. The analysis of the performance of the equaliser is provided in terms of the resulting BER and required computational complexity. Some of the strategies to simplify the resulting equaliser are also discussed.

In Chapter 5, the integration of optical solutions, such as Optical Phase Conjugation (OPC) and Dispersion Managed (DM) links, with Neural Networks (NNs) is explored as a way to reduce the resulting complexity of NNs. Following a brief introduction to Optical Phase Conjugation and dispersion management, the implementation of each of the combined equalisers is described. The performance and resulting computational complexity of the combined equaliser are then compared to that of just NN-based equalisation. It is investigated how these optical solutions affect the computational complexity of NN-based equalisers.

Chapter 6 showcases some of the techniques that were considered for their potential to speed up/enhance learning. Firstly, data pruning, based on the symmetry of QAM constellations, is examined to remove repeated data and accelerate the learning process. Curriculum learning is then investigated, where the training data gradually shifts from easy to difficult, aiding the learning process. Additionally, the use of multi-output networks is explored, enabling the detection of multiple symbols and potentially improving the

performance of the system.

In Chapter 7, to mitigate noise accumulation in photonic NN implementations, a crowd equalisation approach based on committee learning is investigated. Firstly, an introduction to committee learning for crowd equalisation is provided, followed by a comparison of different approaches for combining Neural Networks in crowd equalisation. This is followed by the assessment of the resiliency to the noise of each type of crowd equalisation.

In Conclusion, a summary of key findings is provided. This is then followed by the identification of future directions for further research and development.

1.2 Publications

The results in this thesis include the following publications:

- M. Kamalian-Kopae, A. A. Ali, **K. Nurlybayeva**, A. Ellis, and S. Turitsyn, "Neural Network-Enhanced Optical Phase Conjugation for Nonlinearity Mitigation," in Optical Fiber Communication Conference (OFC) 2022, S. Matsuo, D. Plant, J. Shan Wey, C. Fludger, R. Ryf, and D. Simeonidou, eds., Technical Digest Series (Optica Publishing Group, 2022), paper W2A.38.

- D. A. Ron, **K. Nurlybayeva**, M. Kamalian-Kopae, A. A. Ali, E. Turitsyna, and S. Turitsyn, "On the Impact of the Optical Phase Conjugation on the Computational Complexity of Neural Network-Based Equalisers," in European Conference on Optical Communication (ECOC) 2022, J. Leuthold, C. Harder, B. Offrein, and H. Limberger, eds., Technical Digest Series (Optica Publishing Group, 2022), paper We5.29.

- **K. Nurlybayeva**, D. A. Ron, M. Kamalian-Kopae, E. Turitsyna, and S. Turitsyn, "Noise-Resistant Crowd Equalisation for Optical Communication Systems Based on Machine Learning," in Frontiers in Optics + Laser Science 2022 (FIO, LS), Technical Digest Series (Optica Publishing Group, 2022), paper FM3D.2.

- **K. Nurlybayeva**, D. A. Ron, M. Kamalian-Kopae, E. Turitsyna and S. Turitsyn, "Implementation of Noise-Resistant Crowd Equalisation in Optical Communication Systems with Machine Learning DSP," 2022 Asia Communications and Photonics Conference (ACP), Shenzhen, China, 2022, pp. 753-756, doi: 10.1109/ACP55869.2022.10088872.

Furthermore, the following list of works submitted/in the process of submission:

- **K. Nurlybayeva**, M. Kamalian-Kopae, E. Turitsyna, and S. Turitsyn, "Combining Optical and Digital Compensation: Neural Network-Based Channel Equalisers in Dispersion-Managed Communications System," -submitted.

- **K. Nurlybayeva**, M. Kamalian-Kopae, E. Turitsyna, and S. Turitsyn, "Implementation of Crowd Equalisation in Optical Communication Systems with Machine Learning"- to be submitted.

Chapter 2

Literature Background

Communication systems can be broadly categorised into guided and unguided systems. Guided communication systems, such as optical and microwave systems, transfer signals through a physical medium like coaxial cables or optical fibres. In contrast, unguided systems, such as satellite communication, use free space to transmit signals without needing a physical medium[9]. Optical Communication Systems are a type of communication technology that uses light waves to transmit information from one place to another. They use laser technology to encode information onto a light beam and send it across optical fibres or through the air. This use of optical fibres provides various advantages. Fibre optic systems enable faster transmission speeds than copper-based systems because they use layers of material transparent to the emitting frequency to guide light waves. The signal propagates through a series of total internal reflections, resulting in significantly less signal loss.

Optical communication systems offer several advantages over traditional electrical communication systems, including ([9; 10]):

Bandwidth and Capacity: Compared to microwave systems, which function at carrier frequencies of about 1 GHz, optical communication systems operate at substantially higher frequencies (roughly 200 THz). Because of this higher carrier frequency, optical systems have a large potential bandwidth and can transmit data at bit rates of approximately 1 Tb/s, a significant improvement over microwave systems.

Signal Loss and Distortion: Signals can be transmitted over long distances with little degradation due to optical fibres' ability to transmit light with low dispersion and losses (as low as 0.2 dB/km). In contrast, microwave systems may encounter increased signal loss and interference, especially when operating over large distances.

Long-Haul Applications: Optical communication systems are particularly well-suited for long-haul applications, such as transoceanic communication, due to their high capacity, low loss, and the ability to use wavelength-division multiplexing (WDM) with optical amplifiers to enhance system capacity and reduce overall cost .

Practicality of cables: Cables are lighter and thinner in diameter, making installations

more practical while remaining cost-effective. Additionally, their non-conductive nature enhances security, protecting against remote signal detection.

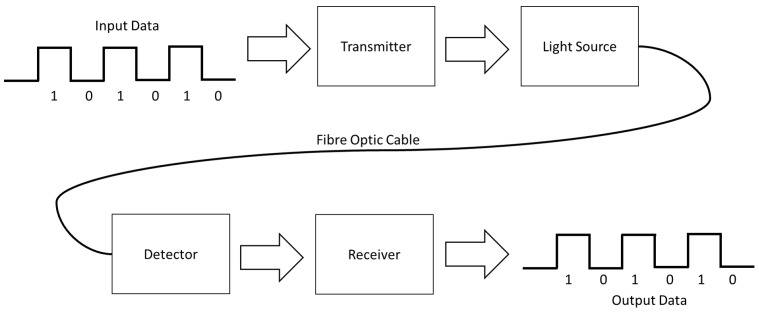

Figure 2.1: Basic fibre optic communication system

.

These advantages play a crucial role in developing the digital economy and are required for modern communication networks. The basic structure of optical communication systems is illustrated in figure 2.1 and consists of the following components:
Optical transmitter - to transform electrical signals into optical signals that can be transmitted over the fibre optic cable.
Optical fibre cable - the medium through which the light signals travel. The cable consists of a core made of glass or plastic, surrounded by a cladding layer and a protective jacket.
Optical amplifier - to amplify the optical signals to compensate for signal loss caused by the distance travelled and other factors.
Optical receiver - to convert the optical signals back into electrical signals to be then processed by the receiving device.

The structure of an optical communication system is designed to provide high-speed and reliable communication while maintaining a low level of signal loss and error rates. The various components are carefully designed and integrated to ensure the system operates effectively and efficiently. Furthermore, depending on the system's requirements, such as the distance of transmission and the bandwidth, the system may require additional components. For example, long haul high bandwidth communication may require wavelength-division multiplexing, optical amplifiers (such as Erbium-doped fibre amplifier (EDFA)), coherent detection, and high-speed infrared photodetectors [10].

2.1 Effects in Optical Fibres

Optical communication differs from standard communication systems because fibres introduce nonlinear effects to the systems, and their characteristics vary with signal power. In the linear channel, increasing signal power improves communication system performance. However, in the fibre channel, increasing signal power enhances nonlinear signal-signal and signal-noise interactions, resulting in significant signal distortions. The signal propagation in the optical communication model in single-mode fibres can be described by the

Nonlinear Schrodinger equation (NLSE) [11]:

$$\frac{\partial E}{\partial z} + \frac{\alpha}{2}E + \frac{j}{2}\frac{\beta_2 \partial^2 E}{\partial t^2} - \frac{j\beta_3}{6}\frac{\partial^3 E}{\partial t^3} = j\gamma|E|^2 E \tag{2.1}$$

Where E is the electrical field as a function of the propagation distance z and time t. α, $\beta_{2,3}$ and γ are, respectively, the effects of fibre loss, chromatic dispersion and Kerr nonlinearity of the optical fibre. This equation can be divided into linear and nonlinear effects of fibre as follows [12]:

$$\hat{L} = -\frac{\alpha}{2}E - \frac{j}{2}\frac{\beta_2 \partial^2 E}{\partial t^2} + \frac{j\beta_3}{6}\frac{\partial^3 E}{\partial t^3} \tag{2.2}$$

$$\hat{N} = j\gamma|E|^2 E \tag{2.3}$$

Where $\hat{L}$ denotes linear effects, and $\hat{N}$ stands for nonlinear.

Equation 2.1 is appropriate to describe optical fibre transmission when transmission along a single polarisation is investigated, such as in intensity-modulation with direct-detection systems [11]. But a coherent transceiver uses advanced Digital Signal Processing (DSP), which makes it possible to identify a dual-polarisation signal and double the system's spectral efficiency. For that, the relations between the two signal polarisations—both linear and non-linear—must be considered. Hence, Eq. 2.1 can be rewritten as follows:

$$\frac{\partial E_x}{\partial z} = -\frac{\alpha}{2}E_x + \frac{j}{2}\frac{\beta_2 \partial^2 E_x}{\partial t^2} - \frac{j\beta_3}{6}\frac{\partial^3 E_x}{\partial t^3} - j\gamma\frac{8}{9}(|E_x|^2 + |E_y|^2)E_x \tag{2.4}$$

$$\frac{\partial E_y}{\partial z} = -\frac{\alpha}{2}E_y + \frac{j}{2}\frac{\beta_2 \partial^2 E_y}{\partial t^2} - \frac{j\beta_3}{6}\frac{\partial^3 E_y}{\partial t^3} - j\gamma\frac{8}{9}(|E_x|^2 + |E_y|^2)E_y \tag{2.5}$$

The impact of fibre attenuation, chromatic dispersion, and Kerr nonlinearity, as shown in the equation 2.1, limit the efficiency of the optical communication system. The following sections briefly explain each impairment introduced by fibre in the equations above.

2.1.1 Fibre attenuation

Due to Rayleigh scattering, absorption of the light and fibre impurities and imperfections, the signal transmitted through the fibre loses a portion of the power[11]. The following equation can be used to determine this power loss:

$$P_T = P_0 e^{-\alpha L} \tag{2.6}$$

Where P_0 is power at an input of a fibre, L is the fibre length, P_T is received power, and α is the attenuation constant that measures total fibre losses from all sources. The power loss is proportional to the length of the fibre, as shown by equation 2.6. As a result, long-

haul transmission demands several optical amplifiers (such as EDFA) to compensate for fibre losses. However, due to the interaction between Amplified Spontaneous Emissions (ASE) noise and nonlinear properties of the fibre, the amplifiers cause parametric noise amplification [13].

Standard single-mode fibre (SSMF) has the minimum attenuation coefficient of approximately 0.19d dB/km at the optical bands C (1530nm to 1565nm) and L (1565nm to 1625nm), which is why these bands are typically used for long-haul communications.

2.1.2 Chromatic Dispersion

Another effect of light propagating through the fibre is called Chromatic Dispersion (CD), which is pulse spreading (in the time domain) due to different frequencies transmitting through the fibre at slightly different speeds. This is due to light sources emitting more than one wavelength, each of which propagates at a different rate when it passes through a glass fibre with a refraction index. Chromatic dispersion is usually represented by group velocity dispersion parameter β_2 or by chromatic dispersion coefficient D_c that can be calculated as:

$$D_c = -2\pi c\beta_2/\lambda_2 \tag{2.7}$$

Where λ is considered wavelength and c is the speed of light. The typical values of chromatic dispersion coefficient D_c for SSMF are around 16-17 ps/km/nm. The chromatic dispersion effects can be compensated by either implementing dispersion compensating techniques or with the linear filter at the receiver. More on this in the section 5.2.

2.1.3 Polarisation Mode Dispersion

The single-mode fibre used in optical communication enables the propagation of two degenerate modes in orthogonal polarisations. However, because of imperfections in the manufacturing process and external stress, practical fibres have asymmetric core shapes along the length of the fibre. This asymmetry can cause mode degeneracy to be broken, resulting in a difference in refractive indices for modes polarised in the x- and y- directions, which is known as modal birefringence [11; 14]. It can be expressed using the following formula:

$$B_m = |n_x - n_y| \tag{2.8}$$

Where n_x and n_y are the mode indices for the orthogonally polarised fibre modes.

The birefringence causes orthogonal signal polarisations to experience different refraction indices, leading to varying propagation speeds and resulting in the random spreading of optical pulses. This effect is called Polarisation-mode dispersion (PMD), which can be a limiting factor in long-haul transmission at higher data rates. Some of the other sources of

PMD include manufacturing variations such as non-uniform stress and geometric irregularities, external factors (such as bending, stretching, and heating), fibre ageing, and issues related to connectors and splices. These imperfections also lead to the randomisation of the polarisation state of signals propagating along the fibre.

The pulse broadening due to PMD can be estimated from the differential time delay between its polarised components, which can be approximated as:

$$\Delta\tau = D_{PMD}\sqrt{L} \tag{2.9}$$

Where $\Delta\tau$ is a differential time delay between the polarisations, D_{PMD}, is the PMD factor of the fibre, and L is the propagation distance. For SSMF, the mean value of D_{PMD} is usually approximately 0.1 $ps/\sqrt{km}$.

While advanced fibre manufacturing minimises PMD impact in modern fibres, older installed fibres may pose challenges for system upgrades to higher data rates. Compensation techniques have been developed to address PMD effects, utilising optical and electronic methods with digital signal processing.

2.1.4 Fibre nonlinearity

In addition to the aforementioned linear impairments, the signal transmission through the optical fibre is impacted by nonlinearities of fibres, such as the Kerr effect. The Kerr effect is defined as the change in refraction index that a medium goes through when it passes through an electric field [15]. It may lead to the highly unpredictable behaviour of the performance of the communication system, as it can induce signal distortion at the receiver and is often considered a significant limitation of data transmission speed and reach. The nonlinear parameter from the Eq. 2.1 can be used to estimate the fibre nonlinearity and calculated as follows:

$$\gamma = \frac{2\pi n_2}{\lambda_0 A_{eff}} \tag{2.10}$$

where λ_0 is the carrier wavelength, n_2 is the nonlinear refractive index, and A_{eff} is the effective mode area of a fibre. The nonlinear effects in the fibre can be generally categorised into signal-noise and signal-signal interaction [11]. The signal-signal nonlinear interference provides a more significant effect on the system's performance; hence, most proposed strategies for nonlinearity mitigation concentrate on solving this distortion. This interference can be further divided into intra-channel, Self-Phase Modulation (SPM), inter-channel Cross-Phase Modulation (XPM) and Four-wave mixing (FWM). Self-phase modulation can be described as a self-induced phase shift that the optical field experiences as it travels through optical fibres. It can be measured as a change in the phase of an optical

field by the following equation:

$$\phi = (n + \tilde{n}_2|E|^2)\frac{2\pi}{\lambda}L \tag{2.11}$$

Where n is the refraction index, $\tilde{n}_2$ is the nonlinear refraction index, and E is the electromagnetic field. In addition to the phase shift, the SPM is responsible for the spectral broadening of the pulses. Cross-phase modulation refers to the nonlinear phase shift caused by interference by another field with a different wavelength, direction or polarisation state [11]. The following equation can define this phase shift:

$$\phi_{NL} = \frac{2\pi}{\lambda}L\tilde{n}_2(|E_1|^2 + 2|E_2|^2) \tag{2.12}$$

Where the first term of the right side of the equation is due to SPM, and the second term is due to XPM. It should also be noted that XPM impacts the nonlinear phase shift twice as much as SPM. In addition to the mentioned interferences, the XPM results in an asymmetric spectral broadening of the pulses. In the case of the four-wave mixing, the new wave is generated as the result of the interaction of three incident waves if the phase-matching conditions are satisfied. Nevertheless, while the effects of SPM can be (partially) mitigated with the use of digital signal processing, XPM and FWM can severely degrade the performance of the optical communication system [16]. In addition to fibre nonlinearities introduced by the Kerr effect, the stimulated nonlinear effects occur in optical fibres, such as stimulated Raman scattering (SRS) and stimulated Brillouin scattering (SBS). However, these effects can be neglected as they are only noticeably significant with a very high optical power that exceeds typical values used in optical transmission [11].

To summarise, nonlinearities in fibre are the dominant factor limiting the performance of optical communication systems. Hence, it is crucial to develop a method that can react to signal degradation and/or compensate for the undesired effects of nonlinearities in order to increase the achievable data rate and/or capacity.

Advanced signal processing techniques can be used to mitigate nonlinearity effects in optical fibre. These techniques help preserve the signal transmission quality and ensure that data is transmitted accurately and reliably. Additionally, the choice of optical fibre and the design of the communication system can also be optimised to reduce the impact of nonlinear effects. The next part will give a general overview of some key parameters of optical communications systems.

2.2 Parameters of Optical Communication Systems

The parameters for an optical communication link can vary depending on the specific system being modelled, but the following are some standard parameters that may be

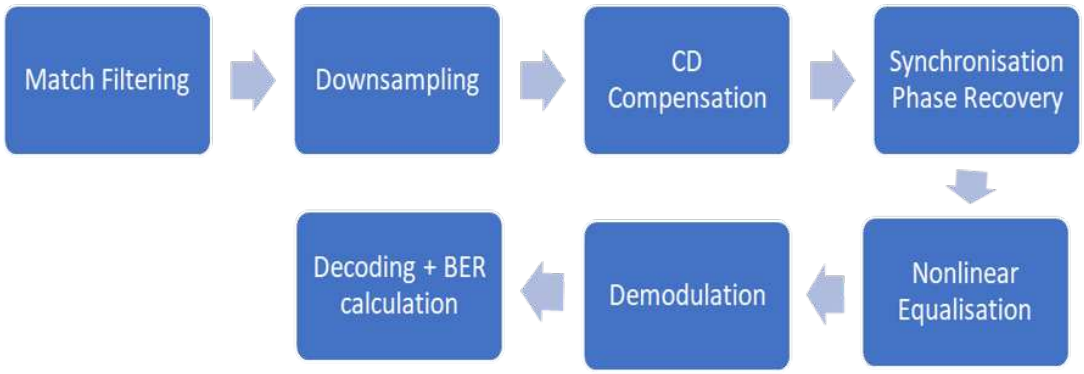

Figure 2.2: Typical DSP processes at the receiver in optical communication link

included in a simulation:

- A number of channels - the number of individual communication paths that can be transmitted simultaneously over a single fibre optic cable.
- Multiplexing technique - the process of combining multiple channels for transmission onto a single fibre optic cable (Time-Division Multiplexing (TDM), Frequency Division Multiplexing (FDM) and Wavelength Division Multiplexing (WDM)).
- Modulation format - the type of modulation used to encrypt data in an optical signal. Examples include quadrature amplitude modulation (QAM), binary phase-shift keying (BPSK), and quadrature phase-shift keying (QPSK).
- Bit rate - the data rate of a system is typically expressed in bits per second (bps).
- Channel bandwidth - the frequency range that the signal is transmitted over.
- Fibre type - the type of optical fibre used in the link can affect the dispersion characteristics of the system.
- Transmission distance - the distance over which the signal is transmitted.
- Optical power -the strength of an optical signal at both the transmitter and receiver.
- DSP at Receiver - various signal processing techniques used at the receiver. More on it is provided in the section below.

It is important to note that the list above is not comprehensive, and depending on the complexity of the system, other parameters may need to be added and adjusted to replicate the real-life scenario.

2.3 Digital Signal Processing at the receiver

Digital Signal Processing plays a crucial role in optical communication links at the receiver. As was already noted, DSP techniques can improve the performance of the receiver by mitigating the linear channel impairments produced during transmission (such as CD and PMD). Figure 2.2 shows an illustration of a typical DSP chain at the receiver.

Match Filtering: In digital communication systems, pulse shaping is a technique

used to modify the pulse waveform of the transmitted signal in order to better manage the signal's bandwidth, reduce spectrum interference, and increase the system's overall efficiency. Furthermore, pulse shaping techniques such as Nyquist pulse shaping or raised cosine filtering are used to mitigate Inter-symbol interference (ISI) by carefully constructing the pulse waveform to reduce symbol overlap and maintain adequate symbol separation at the receiver [9]. The raised cosine pulse shape can be achieved by using matched Rsoot-raised cosine (RRC) filtering at the receiver side in conjunction with RRC pulse shaping at the transmitter side. The transmitter side applies RRC pulse shaping to shape the transmitted signal's pulse waveform according to the desired pulse shape characteristics. The receiver side uses matched RRC filtering to effectively recover the transmitted signal by aligning with the transmitted pulse shape and minimising ISI.

CD Compensation:CD is a type of distortion that remains constant over time and can be counteracted using linear filtering techniques with the inverse CD response [17]. The CD compensation filter can be designed as an all-pass filter with a quadratic phase response. However, because the signal is limited to a specific bandwidth, such as in the case of RRC pulse shaping, the all-pass characteristic of the filter is only necessary within the signal's bandwidth.

Nonlinear Equalisation: Long-haul optical communication systems are susceptible to the detrimental effects of fibre nonlinearity, which can degrade signal quality. Different equalisation methods based on DSP can be used to reduce these effects and improve receiver performance. More information on these methods is provided in section 2.5.

Demodulation: When a transmitted signal is modulated using a specific scheme (such as QPSK, BPSK, or QAM), demodulation at the receiver is performed to restore the original data. This process involves reversing the modulation technique applied at the transmitter to extract the transmitted data. This demodulation operation transforms received symbols into code words represented in bits.

Error Correction and Decoding: Following demodulation, the decoding process occurs, in which these code words are translated into message words (in bits), recovering the original data. Optical communication systems frequently use error correction coding techniques to increase the system's resistance to channel imperfections. These methods raise the data's redundancy, enabling the receiver to identify and fix any transmission problems that may have happened. At receiver-side DSP methods, received symbols or bits are processed using the same error correction code, which removes the redundancy during FEC encoding to detect and correct errors [18].

In hard decision demapping, the received symbols or bits are mapped to their most likely transmitted values based on a specific decision rule, such as selecting the closest symbol in terms of Euclidean or Hamming distance. This method is frequently applied in systems that use Hard-decision Forward error correction (HD-FEC). In soft decision demapping, probabilities or likelihoods are assigned to the received symbols or bits based on the received signal quality and noise characteristics. Soft decision values are used

to estimate the likelihood of each possible transmitted symbol or bit. This method is frequently applied in systems that use Soft-decision Forward error correction (SD-FEC).

Hard decision remapping involves first demodulating each symbol into a code word, followed by decoding this code word. It is characterised by its simple and intuitive methodology [19]. On the contrary, in the soft decision demapping, each symbol is demodulated into a code word, accompanied by a measure of confidence in the accuracy of the mapping. Soft decision demapping has better performance due to its approach to signal interpretation. However, this improved performance comes at the cost of increased complexity and potential error propagation issues.

By employing sophisticated DSP techniques at the receiver, optical communication links can achieve higher data rates, better signal quality, and increased reliability, allowing for large data transmission over long distances.

2.4 Performance measures of optical transmission

Optical communication systems are typically designed to transmit information over long distances using optical fibres. The performance of these systems can be measured using a variety of metrics, some of which include:
Bit Error Rate (BER): is a measure of the number of errors that occur in the transmission of data. It is expressed as a ratio of the number of bits received in error to the total number of bits transmitted.

$$BER[dB] = \log_{10} \frac{number - of - wrong - bits}{number - of - total - bits} \tag{2.13}$$

Error vector magnitude (EVM): is a measure defined as an error vector norm normalised over a signal vector norm. Suppose that the signal comes from the constellation set, c, then the EVM is provided as:

$$EVM = [\frac{|c^{Rx} - c^{Tx}|^2}{|c^{Tx}|^2}]^{1/2} \tag{2.14}$$

Q-Factor: a measure of the quality of a communication system, indicating the separation between signal points in a constellation diagram. It can be calculated from either BER or EVM:

$$Q_{BER} = \sqrt{2} erfc^{-1}(2BER) \tag{2.15}$$

$$Q_{EVM} = 20 \log_{10}(1/EVM) \tag{2.16}$$

Signal-to-Noise Ratio (SNR): is a measurement of the received signal's quality in relation to the system's background noise level.

These metrics play a vital role in optimising the design and performance of opti-

cal communication systems, ensuring reliable data transmission over long distances. In particular, the pre-FEC BER/Q is commonly used to measure the performance of the nonlinear equalisation techniques. This choice is practical for initial performance evaluations because integrating the equalised symbols into the rest of the DSP chain to obtain post-FEC BER/Q is not feasible. The underlying assumption is that if the system operates below a certain pre-FEC threshold, it can achieve error-free operation after FEC decoding. This assumption holds true for HD-FEC systems, as the pre-FEC BER reliably predicts the post-FEC BER. The specific threshold value depends on the overhead and type of encoding employed. Here, an HD-FEC scheme with 7% overhead is considered, resulting in a pre-FEC BER threshold value of $3.8 * 10^{-3}$. This threshold serves as an indicator of the system's ability to achieve reliable communication.

2.5 Existing fibre nonlinearity compensation methods

As mentioned before, nonlinearities in optical fibre significantly limit the capabilities of optical communication systems. Therefore, it is crucial to address the impacts of fibre nonlinearities in order to enhance the attainable data rate and overall capacity. Multiple researchers have investigated various techniques for compensating nonlinear effects in fibres. These techniques can be broadly classified into digital and optical solutions, depending on the way they were implemented.

Digital nonlinearity compensation methods are restricted by the receiver bandwidth, which limits the compensation to intra-channel nonlinear effects [20]. They are primarily based on using digital signal processing to reverse the effect of propagation in the optical fibre. Some examples of such solutions include Digital back-propagation (DBP) [21; 22; 23; 24], Inverse Volterra series transfer functions (IVSTF) [25; 26] and Nonlinear Fourier Transform (NFT) [27; 28]. The complexity of implementing these algorithms is the main pushback for their use for real-time transmission.

In the optical domain, there are several ways to implement nonlinear compensation of fibre effects: 1) by conjugating the signal in the middle of the link (mid-link Optical phase conjugation (OPC)) [29; 30; 31; 32], 2) transmitting a set of mutually Phase-conjugated twin-waves (PCTW) through a nonlinear medium [33; 34; 35], 3) with the fibre backpropagation at the receiver (optical backpropagation). The short survey of how they were used and the problems associated with their implementation for the real-time systems can be found in the table 2.1: As can be seen from the table 2.1, at the moment, each technique has some issues associated with its use for real-time processing of the long-haul optical communication system. It is essential to continue exploring other possible methods for the compensation of nonlinear fibre effects and evaluate their performance compared to the existing methods. My research in this thesis is focused on the use of machine learning-based techniques to account for fibre nonlinearities in optical communication systems. The background of machine learning, its implementation, and an overview of the latest study

Nonlinearity compensation technique	Description	The Problem
Digital Back Propagation (DBP)[21; 22; 23; 24]	Emulates an inverse fibre link propagation using the split-step Fourier method	Very complex, not suitable for real-time implementation
Inverse-Volterra series transfer functions (IVSTF) [25; 26]	Calculates an inverse fibre link propagation using Volterra series transfer functions (allows parallel implementation)	Marginal performance benefit, high computation complexity
Nonlinear Fourier Transform (NFT) [27; 28]	Effectively linearise the channel via nonlinear transform at the transceiver and receiver	Currently, the achievable data rate is below the state-of-art communication systems, high computational complexity
Mid-link Optical Phase Conjugation (OPC) [29; 30; 31; 32]	Performs spectral inversion in the middle of the transmission link	Requires ideal symmetry for the link, reduces the flexibility in an optical routed network
Phase-conjugated twin-waves (PCTW) [33; 34; 35]	Modulation one of the conjugated signals with additional bits or by duplexing the twin waves	Marginal performance benefit or sacrifice in spectral efficiency

Table 2.1: The survey of different methods of mitigating nonlinear effects

into its application to optical communication and fibre nonlinearity compensation are all covered in the next section.

2.6 Implementation of Machine Learning

Machine learning is a branch of artificial intelligence (AI) that focuses on developing models and algorithms that enable computers to learn and make predictions or decisions without being explicitly programmed. Its foundation lies in the concept that systems can acquire knowledge from data, identify patterns, and make intelligent decisions or predictions. Unlike traditional programming, where programmers create explicit instructions to carry out specific tasks, machine learning models are trained using sizeable datasets to learn from examples and generate predictions or decisions based on patterns and statistical analysis. The objective is to create algorithms that automatically learn from experience and improve over time without having to be explicitly designed for each unique task. Machine learning is used in a wide range of fields, including healthcare, autonomous vehicles, image and speech recognition, natural language processing, recommendation systems, fraud detection, and many more. By automating complex tasks, making predictions, and facilitating data-driven intelligent decision-making, machine learning continues to develop and has the potential to completely transform a variety of sectors. In this thesis, the application of machine learning in optical communication is investigated. For that, an overview of the machine learning background is provided in the subsection below.

2.6.1 Machine Learning Background

Machine learning is classified into three types based on the purpose of the learning task: supervised learning, unsupervised learning, and reinforcement learning [36]. Semi-supervised learning refers to the combination of supervised and unsupervised learning algorithms [6]. These ML algorithms are distinguished by their capacity to learn the system's behaviour from previous data and forecast future reactions based on the learnt system model [3]. For this dissertation, the focus is only on supervised learning solutions.

Supervised learning has been used for many applications, such as speech recognition, spam detection and computer vision [3]. The main objective of supervised learning algorithms can be described as giving a set of "historical" inputs to predict an output. It means that some known data consists of inputs and outputs, which are later used to estimate the new outputs based on new inputs [37]. This "labelled data" is called the training set, consisting of N samples of inputs and corresponding outputs. The learning algorithms use it to study the properties of the system by constructing the system response function [36]. Based on the values of the output, supervised learning can be split into solving two categories of problems: Classification and Regression. If the value of the output variable is continuous, then the problem is called Regression. An example of such a problem is predicting the house price based on the provided parameters. However, if the output is discrete, which means as the result of the machine learning algorithm, the output will be classified into a specific category or class; then the problem is described as a Classification problem. As an example, the categorisation of animals into species based

on their characteristics. Furthermore, supervised learning can be broken into two main classes: parametric and non-parametric, based if the number of input parameters is fixed (parametric) or dependent on the training set (non-parametric) [38]. The methods that can be identified as supervised learning algorithms include K-nearest Neighbours, Artificial Neural Networks and Support Vector Machines, among other methods. ML applications in optical communication include resource optimisation by the quality of transmission estimation and prediction and predictive maintenance (fault identification) based on historic traffic or network function patterns [3].

2.6.2 Evaluation of machine learning

When developing a machine learning algorithm, it is essential to consider the design decisions carefully. This includes selecting algorithms, optimising model parameters, and avoiding overfitting/underfitting [36]. A poorly designed system might fail the performance of the machine learning algorithm [39]. This section describes a few techniques to ensure the correct function of the machine learning model.

The overall procedure for evaluation of the machine learning model consists of the following stages: preparing and processing data, feature selection, finding the best algorithm and tuning its parameters, performance validation, and finally, testing the model with the test set. Before the model can be trained, it is essential to prepare the data. Firstly, it may require normalising the available data so the learning procedure will not take an unnecessarily long time. Data preparation may also require adding or removing sample points in case of corrupted data, outliers, or missing values. Then the system's data is divided into training, validation and testing sets [36]. The training set is used for the construction/ training of the model. Typically, around 20-30% of the samples are assigned to the validation set, and this set is applied to the constructed model [4]. The purpose of this is to ensure that the completed model does not only work well on the training data and to optimise the model. If the model's performance on the validation set is not satisfactory, then changes to the model or data must be done, and the training and validation stages are repeated until acceptable "fitting" is achieved [36]. Finally, the test data is applied to the trained model to assess the model performance and to find the desired outcome, which can be predicting outputs, finding patterns, etc. It should be noted that test data is never used during the model construction process.

One of the reasons to divide the data into these sets is to avoid overfitting and underfitting the training data [6]. Overfitting is when the model is too complex for the available dataset; hence, it will fit the training set too closely. As the training data may have had some noisy samples and/or outliers, this results in poor generalisation; for example, it may inaccurately predict any new data points. In contrast, underfitting is when the model is too simple for the available data and fails to fit the training data properly. Figure 2.3 demonstrates the difference between underfitting and overfitting in comparison

to the actual model. A few ways to manage the fitting of the model are to either change

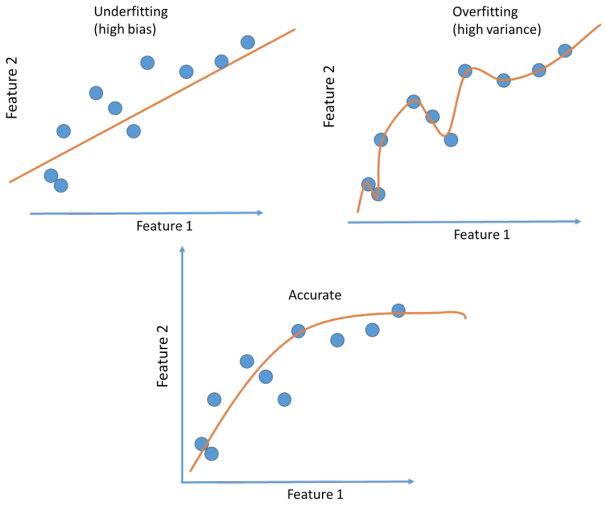

Figure 2.3: Difference between underfitting and overfitting

the complexity (the degrees of freedom of the system), try the other machine learning methods or add the regularisation [40]. Changing the system's complexity is usually done with the step of a feature selection when the features that do not contribute enough to the model are removed or more features are added by combining the existing features. Regularisation becomes valuable when it is difficult or impossible to change the number of features of the system.

As can be seen, outside of the selection of the ML algorithm, there are many design decisions to consider while constructing the model. Tuning of the model is a complex procedure and continuously varies depending on the model and application of the machine learning.

2.6.3 Bayesian Optimisation

In machine learning, hyperparameter optimisation involves selecting optimal parameters to control the learning process of an algorithm. Unlike other parameters that are learned from the data, hyperparameters require tuning to ensure that the model effectively solves the problem by adapting to various data patterns. These hyperparameters are set by the user before the training process and can include variables like the learning rate, number of hidden layers, regularisation strength, and kernel functions.

There are several approaches available that can be used to optimise the hyperparameters of the machine learning model, such as grid search, random search, Bayesian optimisation and gradient-based optimisation. In this work, only Bayesian optimisation is considered.

Bayesian optimisation is a powerful technique designed explicitly for hyperparameter optimisation in machine learning models. Its objective is to find the optimal set of hyperparameters that minimises the model's validation error. This method employs a

probabilistic model to capture the relationship between the hyperparameters and the validation error. The probabilistic model is constantly updated as the model is evaluated with different hyperparameter configurations, providing more accurate predictions of the validation error and guiding the search towards the optimal hyperparameter values. The process is repeated until an appropriate set of hyperparameters is discovered. Compared to other hyperparameter tuning methods, such as grid search or random search, Bayesian optimisation is more efficient and can find better hyperparameters with fewer iterations. However, it is also more computationally expensive and can take longer to run. Overall, Bayesian optimisation is an effective method for increasing the performance of machine learning models.

An overview of the possible issues of optical communication that may be solved with the use of machine learning is provided below.

2.6.4 Machine Learning for optical communication

Machine learning has been considered a possible solution for optical communication systems in the last few years. Some of the reasons that contributed to the increased popularity of ML can be listed as follows:

- Increased system complexity [41]: With coherent detection and digital signal processing, there are more parameters that require tuning to control the system.
- Increased data availability [6]: With many monitors providing information on the optical systems, there is increased availability of the data collected.

Several issues in optical communication have been considered to be partially or fully solved by implementing machine learning, as described in the following table.

Table 2.2 demonstrates the machine learning techniques used to solve the cases of optical communication as found in the existing literature. Despite the found advantages of the use of Machine learning in comparison to the traditional methods [5; 6; 42; 52; 62], overall, several challenges of the realisation of ML for optical communication were found:1) There needs to be an outline of how to design and operate learning-based networks. The choice of an ML algorithm that solves the required problem the best is complex and may require a careful investigation of the different ML approaches and trial and error testing. This aligns with the preceding section, where it was mentioned that extensive tuning might be necessary for the ML model. 2) The research on implementing ML in optical systems is still in its early stages; to date, it has not yet been implemented in actual real-life communication systems. 3) Machine learning requires huge amounts of data; however collecting, processing and transferring this data to the ML model is an open problem. 4) Machine learning in optical communication is a multi-domain problem which requires a complex architecture design.

For the dissertation, nonlinearity mitigation was chosen as the primary focus. This is because nonlinearity in optical fibres affects the transmission's performance in terms of

Application	Description	Relevant ML technique
Estimation of Quality of Transmission (QoT)	Predict QoT of the unestablished lightpaths or monitor the QoT of current lightpath to find the faults	Random forests classifier (RF) [42], Support Vector Machine (SVM) [43], Regression [44], Case-Based Reasoning (CBR) [45]
Optical amplification control	Reconfigure the network devices as any changes in the WDM system require EFDA gain to adjust to rebalance output powers	Neural Networks (NN) [46], CBR [46], Kernelised linear regression [47]
Modulation format recognition and automatically identify the modulation format from the features of the incoming optical signal	Principal Component Analysis (PCA) [48], NN [49], SVM [50], Clustering K-means [51]	
Nonlinearity mitigation	React to signal degradation and/or compensate undesired nonlinearities	Bayesian filtering [4], NN [52; 53; 54; 55], k-nearest neighbours [56], N-SVM [57], binary SVM [58], Clustering K-means [59]
Optical Performance Monitoring	Estimate transmission parameters for future use	NNs [60], SVM [50], Gaussian Processes [61]

Table 2.2: The overview of applications of Machine Learning in optical communications

BER and quality factor (Q-factor). The survey of the latest achievements of ML-based nonlinear equalisers is presented in the section below.

2.6.5 Machine Learning for nonlinearity compensation

Machine learning has primarily been used at the receiver and is considered to be a promising way to mitigate nonlinear effects. The following list outlines the possible benefits of this approach over the ones currently in use [3; 4; 42; 63]:

- The complex nonlinear behaviour of optical fibres can be modelled using machine learning methods, which can reduce the nonlinear effects caused by both deterministic nonlinearities and stochastic nonlinear signal ASE noise interactions.
- Machine learning algorithms are able to predict the nonlinear behaviour of optical fibres solely from the input-output data. As a result, it is not necessary to have a thorough knowledge of the optical link parameters, which can be challenging in dynamic optical systems.

- Dynamic optical systems can be handled, as the models can be trained on data from dynamic optical systems, allowing them to adjust to changes in the system over time. This is in contrast to traditional methods, which might need to re-tune or re-optimise settings when the system changes.
- When compared to conventional methods that are based on heuristics or simple models, machine learning algorithms have the potential to achieve high levels of accuracy in nonlinear mitigation.

Numerous ML-based equalisers have been proposed over the course of years of research, including support vector machines (SVM), K-means clustering, the K-nearest neighbour algorithm, Neural Networks and many others [3; 5; 6; 24; 40; 62]. However, defining which ML methods may provide the best results can be challenging. In optical transmission-related issues, achieving a minimum of 99% accuracy in determining the transmitted bits is often necessary for state-of-the-art transponders operating at pre-Forward Error Correction (FEC) bit error rates (BER). This means that the ML-based equalisers have to be extremely accurate. Furthermore, for real-time implementation of the equaliser, it is crucial to minimise latency to the greatest extent possible. Striking a balance between high accuracy and low latency poses a significant difficulty in developing ML-based equalisers for optical communication systems. Moreover, striving to achieve such a high accuracy requirement in machine learning can give rise to multiple challenges, including the risk of overfitting. It is also crucial to evaluate the performance of different machine learning algorithms and compare them. The evaluation process should consider several factors, including the computational complexity, training time, and the achieved performance of machine learning models. This evaluation can help to identify the most promising machine learning techniques for reducing nonlinear effects in optical communication systems.

In addition, further research should also focus on developing new approaches to the implementation of machine learning algorithms that can provide better performance than existing techniques while also being computationally efficient. This would require a combination of domain expertise in optical communication systems and machine learning, as well as access to appropriate data sets and computing resources.

Chapter 3

Support Vector Machine for nonlinearity mitigation

The Support Vector Machine is a binary classification machine learning technique introduced by Vapnik [64]. Though it is primarily used for classification, it can be modified to perform regression tasks. SVMs generally can handle complex decision boundaries and usually work well with high dimensional data; hence, they are a good starting choice as a possible solution for fibre nonlinearity compensation. Furthermore, SVM is generally a low-complexity algorithm, which means that it could be suitable for real-time implementations [65]. However, the algorithm's complexity and performance can significantly vary based on how SVM is implemented. The following section will provide a brief background of SVM and a description of parameters that can be tuned to improve the learning algorithm's performance.

3.1 Support Vector Machine

The fundamental concept of SVM is to find an optimal hyperplane that effectively separates the data points of different classes with the widest margin. This hyperplane acts as the decision boundary, and the data points, which are the closest to it and used to calculate the margin, are known as support vectors(see Fig. 3.1). A kernel function is introduced for the linearly inseparable data, so it maps the data to a high-dimensional feature space, where the separation can be solved easily. The common kernel function for the SVM is the Gaussian radial basis function (RBF) [66]. The principle of SVM for binary classification can be described with the following: for $[xy]$ input training data, where x is MxN matrix representing the features, and y is $Mx1$ matrix representing class labels, the write function of nonlinear SVM can be written as follows:

$$\min_{\xi,\omega,b} \|\omega\|^2 + C\sum_{i=1}^{N} \xi_i \tag{3.1}$$

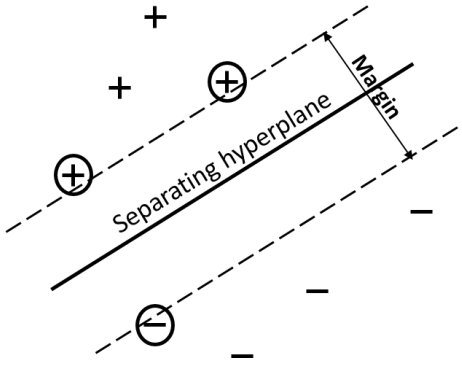

Figure 3.1: The concept behind Support Vector Machine (SVM)

Such that:

$$y_i(\vec{\omega}^T x_i + b) \geq 1 - \xi_i \tag{3.2}$$

$$\xi_i \geq 0 \qquad i = 1, ..., M \tag{3.3}$$

Where ω is a vector perpendicular to the hyperplane, C is a penalty factor that controls the margin between classes, ξ_i is the slack variable, and b is the bias term. The parameter C has been introduced into the system to avoid overfitting of a model, so in case of some misclassification, the data can be "ignored", which can lead to a better fit. The optimisation problem can be reformulated by introducing Lagrange multipliers α_n for each constraint, and by using Karush-Kuhn-Tucker (KKT) complementary conditions, the coefficient of SVM α can be derived [66]. The prediction of the output of each data point (z_k) depends on kernels evaluated at a subset of the training data points and is defined as follows [64]:

$$z_k = sign(\sum_{n=1}^{N}(\alpha_n \vec{y_n} K(x_k, x_n) + b)) \tag{3.4}$$

Where $K(x_k, x_n)$ is a Kernel function.

In the SVM, the following parameters are that are considered important, as they can significantly affect the performance of the algorithm:

- Kernel Function –defines the shape of the hyperplane. Based on the function that describes its shape, it can be linear, Gaussian, sigmoid or polynomial.
- Kernel Scale - parameter that is used in the kernel function, such as g in the Gaussian kernel. For the Gaussian kernel, if the value of the kernel scale is high, then features vary more smoothly, and on the opposite, a small kernel scale will result in features varying less smoothly.
- Box Constraint - regulates the maximum penalty imposed on margin-violating observations, which helps to prevent overfitting [67]. High values of box constraint

prioritise the accuracy of prediction, while low values prioritise the algorithm's simplicity (hence, speed of training).

The optimisation of these parameters can be done either manually by testing different values of parameters or by using such functions as Bayesian optimisation, which selects optimal parameters that minimise the cross-validation loss.

As SVM is a binary classifier, the N QAM system requires combining multiple SVMs using one of the methods of multi-class classification. There are several strategies for classifying the SVM algorithm into multiple classes. Below are the most widely used strategies for multiclass classification:

- *One vs Rest* – a received constellation point would belong to a certain cluster if that cluster accepted the received symbol and other clusters rejected it. However, the imbalanced distribution of classes can cause the classifier to be overly biased towards the majority class, leading to poor performance in identifying the minority class. This is especially problematic when the minority class is of particular interest or importance.

- *One vs One* - generates an optimal decision function between every two different categories of training sets [68]. The number of samples per training is relatively small; hence, the training speed of a single decision surface is faster, and the precision is higher.

- *Binary Encoding* - the principle is that starting from the root node, the category contained in the node is divided into two subclasses. Then the two subclasses are further divided, etc, until each subclass contains only one category, thus obtaining a binary tree [68]. The SVM classifier is trained in each decision node of the binary tree to realise classification. An N QAM communication system would require $N-1$ SVM classifiers for the training and, depending on the implementation, an even less number of SVMs when testing.

The difference between these methods is that *One vs Rest* requires less classification; hence, it is a faster algorithm. However, the downfall is that the classes may be imbalanced, which would affect the accuracy of the classification. On the contrary, *One vs One* is less sensitive to imbalance but would require more classification, which means it is more accurate but requires more computation. More details on the ways multiclass SVM can be implemented and how each strategy affects the transmission's performance and estimated computational complexity are provided in the table 3.1 [68].

The *One vs One* multi-class method was used to implement SVM for use in these simulations. Additionally, the Gaussian function was selected as the kernel function following a number of sets of experiments and is described with the following equation:

$$K(x_k, x_n) = \exp\left(-g||x_k - x_n||^2\right) \tag{3.5}$$

Method	Number of required SVMs (training)	Number of required SVMs (testing)
One vs one	$N \times (N-1)/2$	$N \times (N-1)/2$
One vs all	N	N
BE-Complete binary tree	$N-1$	$\log_2(N)$
BE-Partial binary tree	$N-1$	$(N+1)/2 - 1/N)$
BE- Constellations rows and columns	$N-1$	$\log_2(N)$
BE-In-phase and quadrature components	$N-1$	$\log_2(N)$ or $(N+1)/2 - 1/N$

Table 3.1: Different models of multi-class SVM, where BE is Binary Encoding

The prediction of the received symbols is determined by classifying them into the N groups as a result of the application of SVM. The SVM architecture is displayed in Figure 3.2. At the SVM's input, delay taps have been used to take the channel memory effect into account. This is done to capture the sequential relationships between consecutive symbols. This parameter controls the length of the input vector to SVM-based equaliser, which, in turn, affects the Computational Complexity (CC) of the equaliser and the processing delay in the system. Additionally, the received symbols were split into real and imaginary values, and an SVM feature vector was created with the help of the information from both polarisations.

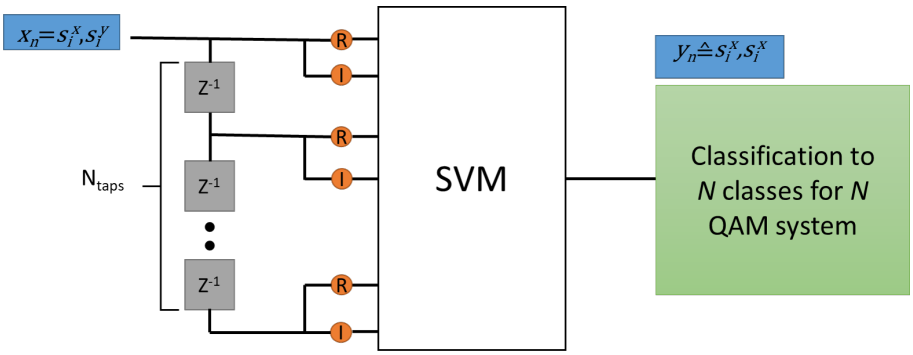

Figure 3.2: The SVM architecture, where s_i^x is the ith symbol of polarisation X, N_{taps} is the number of delay taps

3.2 Support Vector Regression

The Support Vector Regression is a regression type of algorithm that is based on SVM but with the following differences:

- As the output is a real number, it becomes very difficult to predict the information

at hand, which has infinite possibilities.

- A margin of tolerance (ϵ) plays a significant role in the outcome.
- The algorithm is more complicated.

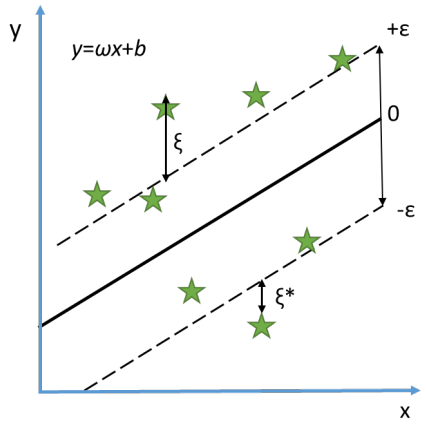

Figure 3.3: The concept behind Support Vector Regression (SVR)

The training of this algorithm can be formulated as follows [64]:

$$\min_{\xi,\vec{\omega},b} \frac{1}{2}||\vec{\omega}||^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \tag{3.6}$$

Subject to constraints:

$$y_i - \vec{\omega}x_i - b \leq \epsilon + \xi_i \tag{3.7}$$

$$\vec{\omega}x_i + b - y_i \leq \epsilon + \xi_i \tag{3.8}$$

$$\xi_i, \xi_i^* \geq 0 \qquad i = 1, ..., M \tag{3.9}$$

Where $\vec{\omega}$ is a vector perpendicular to the hyperplane, C is a penalty factor that controls the margin between classes, ξ_i is the slack variable, and b is the bias term. The prediction of the output of the new data points can then be calculated as follows:

$$z_k = \sum_{n=1}^{N}((\alpha_n - \alpha_n^*) \cdot K(x_k, x_n) + b) \tag{3.10}$$

where $K(x_k, x_n) = \exp(-g||x_k - x_n||^2)$ is a Gaussian kernel function. As an output of SVR is a real number, the nonlinear equalisation of the N QAM system can be implemented by combining two SVRs: one to predict the real parts of the output, and one- the imaginary, as seen in Fig. 3.4. This structure plays a significant role in determining the computational complexity of SVR-based equalisers. It should also be noted that different configurations of combining SVRs can be used, for example, one SVR to predict the magnitude and second, the phase of the output. The input vector to the SVR-based equaliser was similarly combined by using the information of the N_{taps} consecutive symbols of both

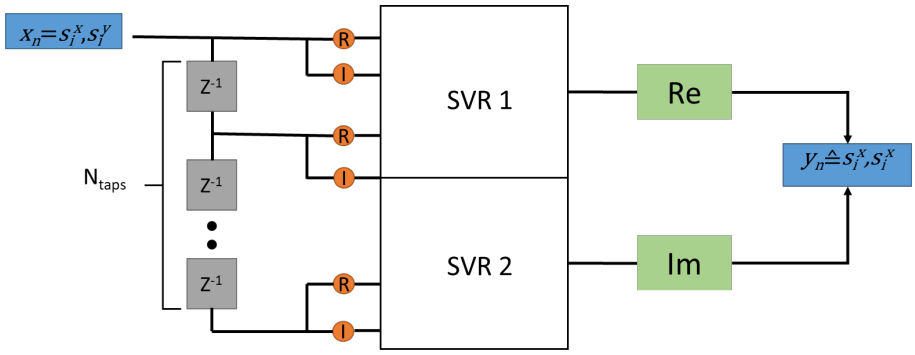

Figure 3.4: The SVR architecture, where s_i^x is the ith symbol of polarisation X, N_{taps} is the number of delay taps

polarisations, split into real and imaginary values.

3.3 The Simulation setup for the testing of the SVM and SVR-based equalisers

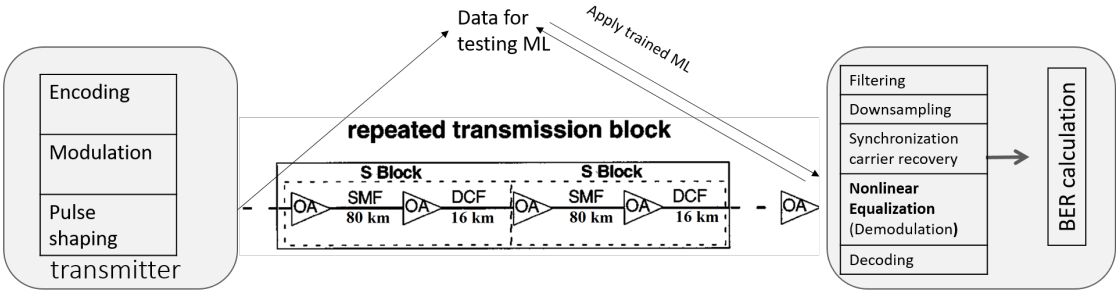

Figure 3.5: The Scheme of the simulated transmission link used for the SVM/SVR experiments.

To assess the performance of the proposed implementation of SVM and SVR-based equalisers for compensating fibre nonlinearities, a simulation of the transmission link was conducted (see Fig. 3.5). A single channel dual polarised transmission at 9 GBaud using 16 QAM signal has been considered with a roll-off factor of Raised-cosine filter set at 0.001 and an oversampling factor of 6. The number of spans has been set at 20, corresponding to 80 km of standard single mode fibre (SSMF) and 16 km of dispersion compensating fibre (DCF). To test the performance of SVM and SVR for the nonlinear equalisation, a DM communication system with fully inline chromatic compensation was chosen with the following parameters (see Table 3.2). The transmitter and receiver provided the data for training and testing of the selected models (as shown in Fig. 3.5). The k-fold selection was used to divide this data into training and validation stages. With the help of Bayesian optimisation, which chooses the best possible parameters by minimising the cross-validation loss, all of the values for the SVM and SVR hyperparameters that were not specified have been discovered for each experiment. In contrast to the frequently used means squared

Launch power Range	-3:2 dBm	Number of Spans	20
SMF span length	80km	DCF span length	16km
SMF dispersion	17 ps/nm.km	DCF dispersion	-167 ps/nm.km
SMF dispersion slope	-13.1	DCF dispersion slope	-15.5
SMF effective area	$95\mu m^2$	DCF effective area	$27\mu m^2$
SMF n2	30 nm^2/W	DCF n2	30nm^2/W

Table 3.2: The parameters of the transmission link used for the experiments with SVM and SVR

error, BER was used as a performance metric for the Bayesian optimisation. This is because BER is the parameter that is of interest to decrease, and optimising by MSE might not always result in improved BER. After the optimised algorithms were applied to the testing dataset, the performance of SVM and SVR was compared. The performance of the algorithms for nonlinearity mitigation was assessed by evaluating BER and calculating the computational complexity of the algorithms.

3.3.1 Results with Support Vector Machine

Firstly, it was essential to identify the optimum dimensions of the input (feature) vector to SVM, which is related to the channel memory. Therefore, it was essential to examine how SVM performs when only data for a single polarisation is used when using a different amount of delay taps. For that, a Bayesian optimisation was applied to define the ultimate parameters of the SVM for each launch power at each value of the delay taps. Figure 3.6 shows the resulted BER of this experiment as a function of the launched signal power. As seen from this figure, using SVM as a nonlinear equaliser has provided some improvement in BER, but not enough to reach below the HD-FEC threshold. When comparing the results of the one delay tap versus three delay taps, it can be seen that BER has not improved as the number of delay taps has increased. This implies no correlation between adjacent symbols, suggesting that the effective channel memory is small. This is due to the inline dispersion management reducing the channel memory. Additionally, the optimum launch power has not been changed, and it can be seen that an increase in launch power leads to a more percentage improvement in BER, see Fig. 3.7. Hence, it can be concluded that with the improvement in the high power (nonlinear-regime) area, SVM is tackling the nonlinearity problem.

Figure 3.6 also demonstrates the results of the equalisation with SVM when the data from both polarisations is used. As can be seen from the figure, the equalisation results are similar to the case of single polarisation, except there is more BER improvement as the graph touches the HD-FEC threshold. Both tests showed that raising the number of

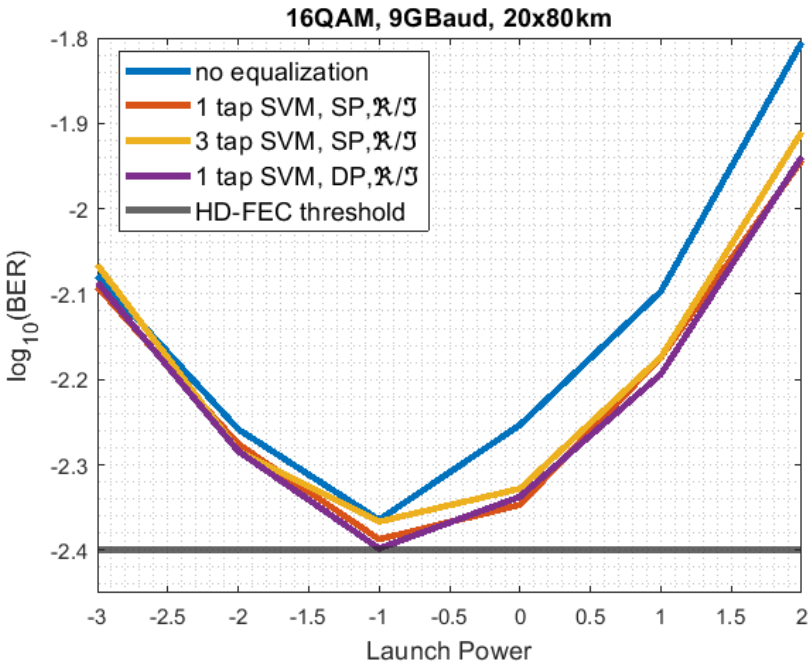

Figure 3.6: The BER results of SVM for single and dual polarisation using ℜ/ℑ

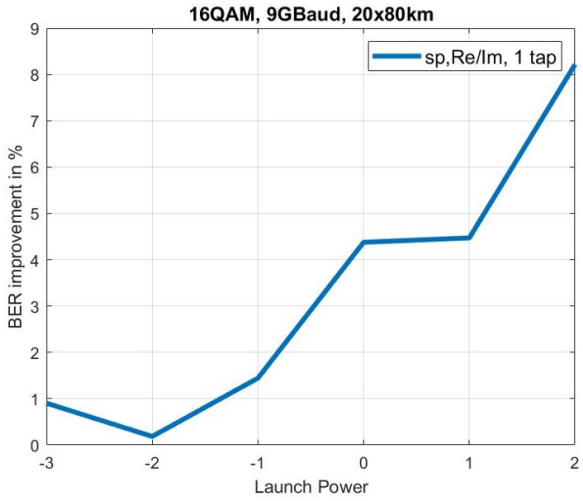

Figure 3.7: The percentage improvement of SVM for single polarisation as a function of Launch Power using ℜ/ℑ

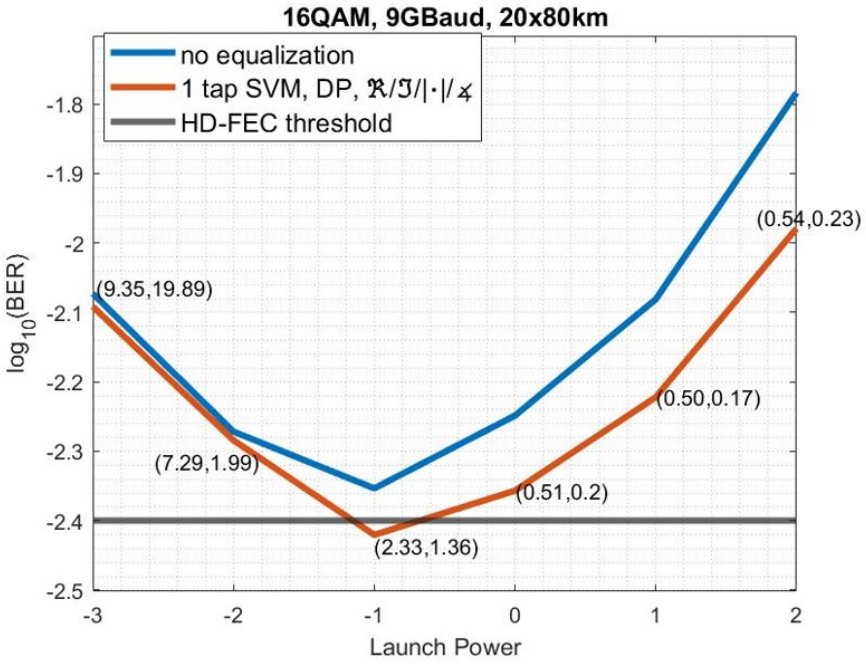

Figure 3.8: The results of SVM for dual polarisation using $\Re/\Im/|\cdot|/\angle$

delay taps has little effect on BER. Additionally, using data from both polarisations to run simulations produces results that are only slightly better than using data from a single polarisation, necessitating the need for an alternative method of feature generation. It was decided to add magnitude and angle to the received symbols in addition to the real and imaginary components. Figure 3.8 demonstrates the results of the updated implementation of SVM. Adding more features, such as magnitude $|\cdot|$ and angle $\angle$ of the received symbols alongside used $\Re$ and $\Im$ has resulted in a slightly better BER performance of SVM.

At the optimum power, the BER, obtained by the SVM equaliser, crosses the HD-FEC threshold and provides around 0.06 dB improvement. This improvement results from SVM's new decision boundaries, which are different from the straight lines coming from the maximum likelihood detection. Figure 3.9 demonstrates the resulting decision boundaries by SVM at 2 dBm launch power. The decision boundaries are round, as the Gaussian kernel function was used. Some of the symmetry can be seen with these decision boundaries, which can be used to reduce the computational complexity of SVM. From there, it can be said that implementing classification and redrawing boundaries seems not to be very effective in reducing the BER. This is because the placement of the boundary line depends not only on the symbol of interest but also on preceding and subsequent symbols. Consequently, redrawing the boundaries involves a process of averaging across various potential realisations of symbols in the vicinity of the symbol of interest.

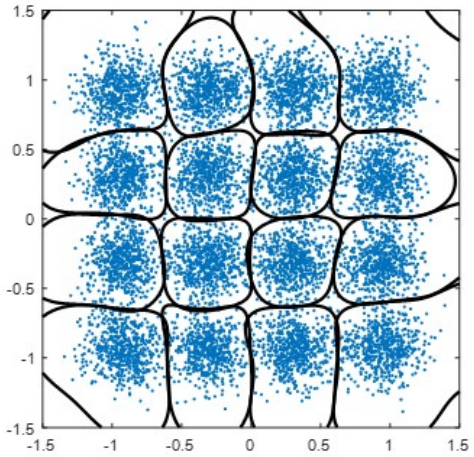

Figure 3.9: Achieved decision boundary by SVM at the optimum launch power

3.3.2 Results with Support Vector Regression

A similar set of experiments with different feature vector structures have been performed to test SVR for fibre nonlinearity mitigation. Figure 3.10 demonstrates the current best result of nonlinear equalisation using SVR. This result was obtained using $\Re/\Im/|\cdot|/\angle$ from single polarisation for the feature vector to SVR. SVR improves the performance by 0.02 dB in BER, but the resulting BER is above the HD-FEC threshold. Additionally, the experiments revealed that adding more delay taps had not improved the system, proving that this system has little/no channel memory. The main distinction between SVR and SVM is that in regression, a maximum likelihood detection is implemented, which results in decision bounds that are linear. This is because SVR, as a regression ML, relocates the symbols, helping to invert the effects of nonlinear distortions. Figure 3.11 shows the final constellation that was found after equalising using SVR. From there, it can be seen that SVR produces window constellations, just like other regression tasks of a similar nature [8]. Furthermore, there are a few missing points at the angle of π because $\angle$ was one of the input features. This is because there is a discontinuity in phase on the negative real axis.

3.4 Computational Complexity Analysis of SVM and SVR-based Equalisers

It is essential to conduct a computational complexity analysis to determine whether machine learning-based NLEs can be practical for real-time implementation. The difference in calculating required computational power with existing techniques, such as Digital back-propagation, is that ML-NLE presents a complexity that does not depend on the link parameters. Instead, it depends on some signal parameters, such as the number of constellation points and subcarriers [63]. Subsequently, the computational complexity of a receiver based on the proposed SVM architecture was compared with a receiver that used

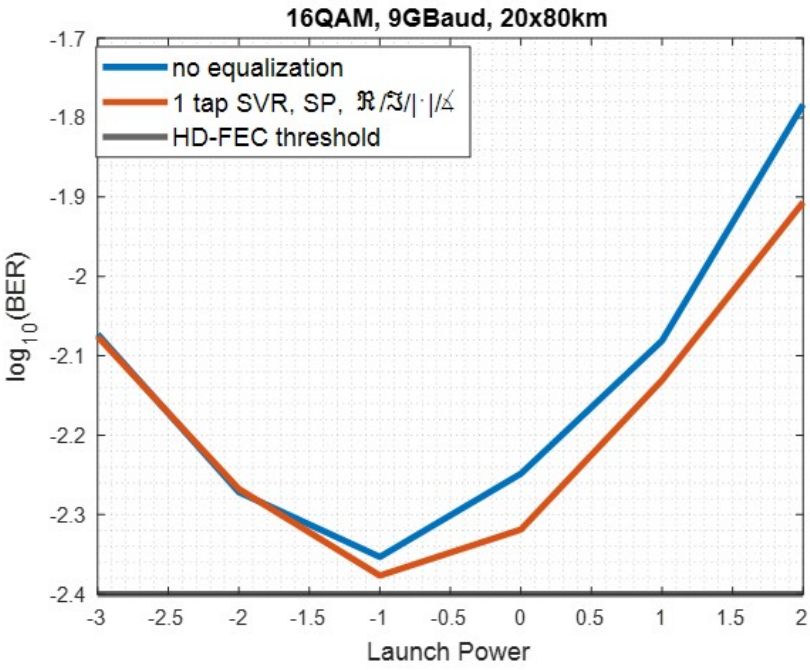

Figure 3.10: Best results of SVR for single polarisation using $\Re/\Im/|\cdot|/\angle$

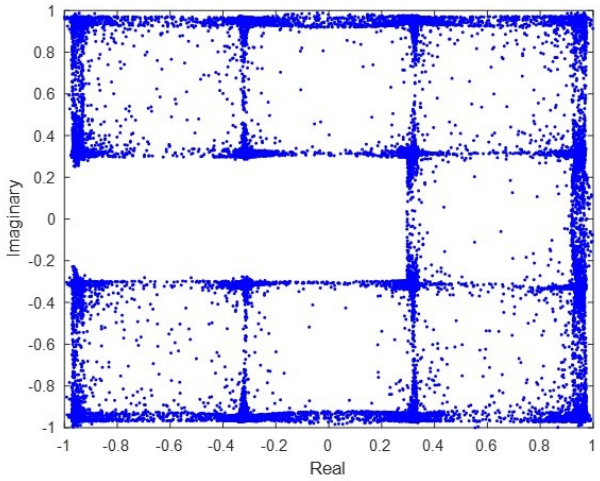

Figure 3.11: Achieved constellations by SVR at the optimum launch power

the SVR. The comparison was measured in the total number of real multiplications per transmitted bit required by each nonlinear compensation scheme.

As previously discussed, equalising a N-QAM signal into N classes requires the implementation of multiple SVMs. Consequently, the overall complexity of such an equaliser can be expressed by the following equation:

$$CC = \sum_{j-1}^{M} (CC_j) \tag{3.11}$$

where CCj is the computational complexity of a single SVM and M is the number of SVMs. The multi-class classification strategy will affect M. The expected number of SVMs needed for both the training and testing phases for each of the strategies is shown in Table 3.1. In this experiment, a *OnevsOne* strategy for multi-class classification was selected to equalise the signal of a 16 QAM communication system. This means that a total of $M = 16 * (16 - 1) = 120$ SVMs are needed for both the training and testing phases of equalisation.

The SVM model can be trained offline and loaded into the memory. To estimate the required number of multiplications per transmitted bit for the testing stage of a single SVM, it is required to look at Eq. (3.4). The computational complexity of this equation depends on the number of support vectors obtained during the training stage and the chosen kernel function. The computational complexity of the kernel function is fixed after determining the optimum dimension of the feature vector. Therefore, the CC of SVM-NLE depends on the number of support vectors and the size of the feature vector. For the best-achieved implementation of SVM, the dimension of the feature vector is $d = 2 \times 4 \times N_{del}$. If n is the number of support vectors, solving Gaussian kernel n times would require $d + 1$ multiplications and $2d - 1$ additions. Hence, in theory, $n \times (d + 1)$ multiplications and $(n - 1) \times (2d - 1)$ additions are required for the decision of every received symbol. The estimated calculation complexity for the best-achieved SVM implementation (using $\Re/\Im/|\cdot|\angle$ from both polarisations for the feature vector) is shown in Fig. 3.12. This image illustrates that the required number of support vectors increases with launch power, which corresponds to the increase in the nonlinearity. This is because nonlinear data requires more complex shapes of decision boundaries. As a result, solving nonlinearity demands more processing power.

Similar to SVM, the equation Eq. (3.10 was examined for the prediction of new data points in order to determine the computing cost of the SVR-NLE. It can be seen that this equation is very similar to the SVM one, as it also depends on the kernel function and support vectors. However, while the same kernel function was used, the number of support vectors required for a single SVR is higher than SVM. Nevertheless, when comparing the CC between our SVM and SVR implementations, SVR-NLE is generally less complicated because 120 SVMs and only 2 SVRs are required to mitigate nonlinearities.

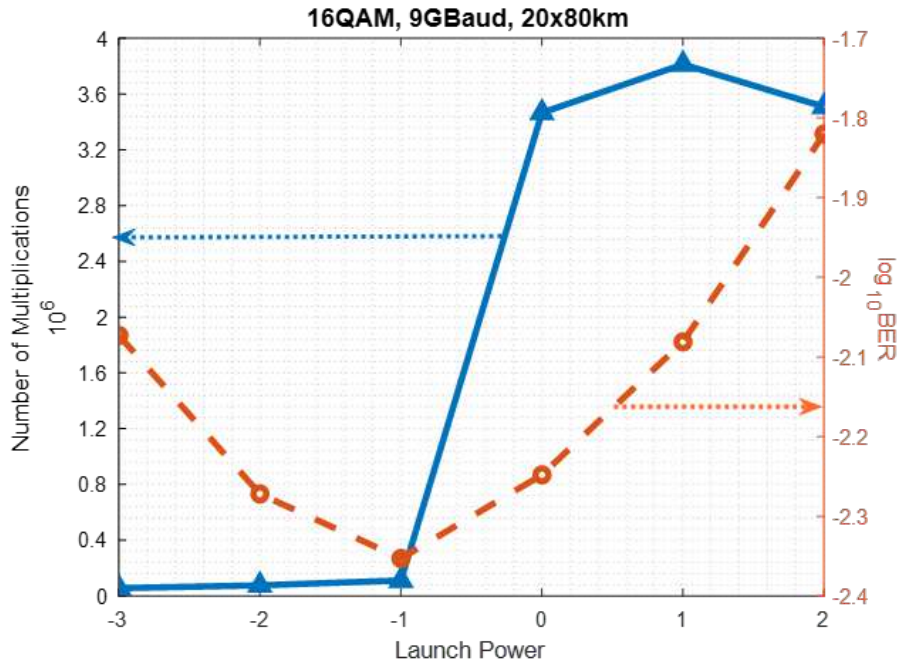

Figure 3.12: Resulting computational complexity by SVM (dual polarisation)

3.5 Conclusion

SVM-based and SVR-based equalisers were designed for nonlinearity mitigation in the receiver of a 16-QAM Dispersion managed communication system at 9 GBaud with a total (SSMF) transmission length of 1600 km. To accomplish this, optimal configurations for SVM and SVR have been investigated. SVM-NLE enabled a marginal increase in BER at the optimum power in simulations, allowing it to cross the HD-FEC threshold. SVR-NLE has hardly improved BER and is still above the HD-FEC cutoff. It was discovered that the current implementation of SVR-NLE would require less computational power than SVM-NLE. It has been seen that the computational complexity of both machine learning algorithms relies on the number of features and support vectors. Hence, reducing the number of support vectors or using a different multi-class classification method are two strategies for reducing the computational complexity of the obtained nonlinearity compensation methods. However, the SVM and SVR's optimised (complex) structure only demonstrated a marginal performance improvement at a high computational complexity cost. This means that there is no need to investigate the reduced complexity versions of the SVM and SVR since they would not result in a better BER improvement, especially since they hardly cross the HD-FEC threshold. Hence, it was decided to investigate different types of machine learning that are more known for providing a higher accuracy in predictions- Neural Networks.

Chapter 4

Neural networks

Previous research has shown that Artificial Neural Networks (ANN) demonstrate impressive performances when used as equalisers to mitigate nonlinear impairments in modern optical communication systems [4; 8; 63]. Despite its remarkable performance, such equalisers have a high processing cost when compared to other methods like digital backpropagation (DBP) [62; 69] This, in turn, limits the real-time implementation of Neural Networks for fibre nonlinearity mitigation. As a result, it's essential to maintain the equaliser at a low level of complexity while keeping adequate performance.

In this chapter, firstly, Complex-valued Multilayer Perceptron will be explained, followed by a demonstration of how it was implemented and how the optimal hyperparameters were found. Then the performance of the chosen equaliser is demonstrated for the fibre nonlinearity mitigation in terms of BER and required computational complexity.

4.1 Multilayer Perceptron

One of the most commonly used and straightforward machine learning techniques for equalising communication signals is the multilayer perceptron (MLP). This feed-forward neural network (NN) comprises multiple layers containing neurons and is represented by nonlinear activation functions with linear connections between the layers. A weighted total of the inputs is performed by the neurons in each layer of the MLP, and an activation function is then applied to create an output, see Fig. 4.1. The subsequent layer receives each neuron's output after that (Fig. 4.2). To reduce the discrepancy between the actual output and the targeted output, the weights of the connections between the neurons are modified during training using an optimisation algorithm, such as backpropagation.

There are two primary reasons why the MLP is a desirable equalisation method for fibre nonlinearity mitigation.

- the simple structure of MLP with relatively low computational complexity (in comparison to other Deep NNs) [62]. This is primarily caused by the fact that the numerical implementation of the MLP's nonlinear activation functions is built on a

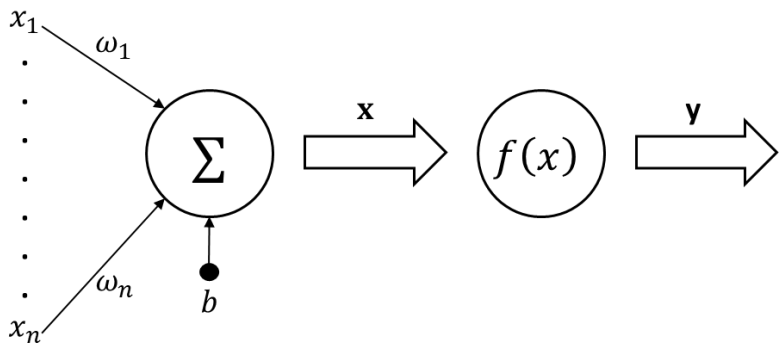

Figure 4.1: A structure of simple neuron with vector input, where $x_{1,...n}$ is the input vector, y is the output, $\omega_{1,...n}$ are the weights and b is the bias .

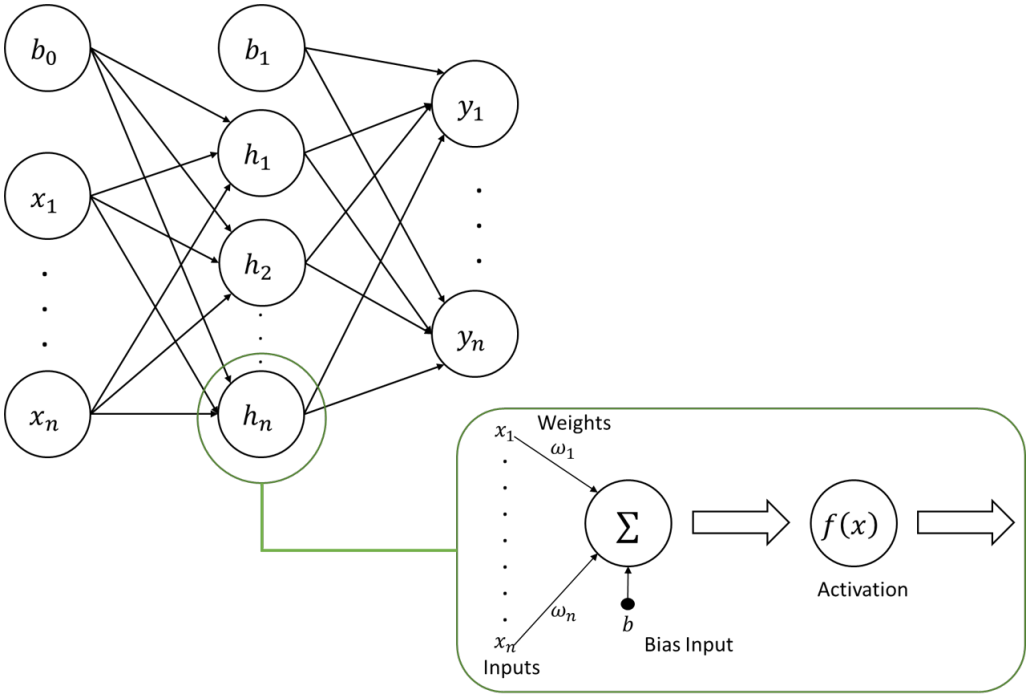

Figure 4.2: An example of the configuration of a multilayer neural network.

look-up table with a minimal computational load.

- With the help of photonic devices, which are inherently quick and power-efficient, several methods exist to implement an MLP in the optical domain. The weighted accumulation and nonlinear function are the fundamental components of an MLP. While the latter is more complex, there are proposals with experimental proof of their viability [70; 71]. The former can be done using optical components [71; 72].

The number of layers and neurons in each layer for such a simple neural network decides the computational complexity, which in turn controls other crucial characteristics of the equaliser like power consumption, needed memory and induced processing delay. However, MLP can experience overfitting and requires a substantial quantity of training data to yield reliable results.

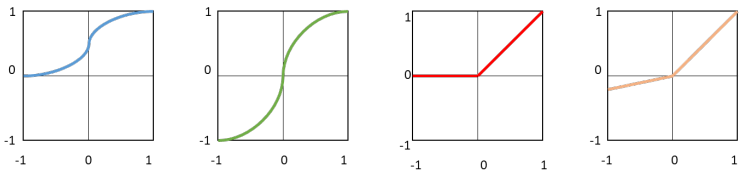

Figure 4.3: Various forms of nonlinear activation functions.

The following list of parameters plays an important role, as it can significantly affect the performance of the neural networks.

Activation function - a mathematical function that defines the node's output in response to an input or set of inputs. Some of the most popular activation functions are linear, hyperbolic tangent (tanh), rectified linear unit (relu), sigmoid, etc. Figure 4.3 shows the shape and equations of some of the popular nonlinear activation functions [73]

Loss function- a mathematical function that calculates the discrepancy between estimated and actual outputs for a given input. Frequently used functions are Mean Squared Error (MSE), Binary Cross-Entropy, Categorical Cross-Entropy, etc. The loss function should be carefully selected because it reflects both the nature of the neural network's output and the problem it is trying to address.

Optimiser - algorithm to adjust the weights and biases during training to reduce the loss function. Commonly used algorithms include Stochastic Gradient Descent (SGD), Adam, RMSprop, etc.

Learning rate (lr) - a hyperparameter that regulates the optimisation algorithm's step size when training a neural network. It decides how much the neurons' weights are updated during each iteration of the optimisation algorithm. Small values of lr cause slow convergence and extended training times, while large values of lr can cause the minimum loss function to overshoot.

Number of epochs - the total number of times the complete training dataset is processed during training. This has a significant impact on model underfitting/ overfitting. It can be adjusted with the use of early-stopping -to stop training when the performance of the model on the validation dataset stops improving.

Batch size - a fixed-size part of the training data, which is used by the optimisation algorithm in one forward/ backward pass to update the weights and biases. The choice of batch size has an impact on the optimisation's training time and convergence stability.

The choice of activation function in a neural network depends on various factors, including the characteristics of the problem, the architecture of the network, and the properties of the activation functions themselves. Sigmoid or tanh activations functions can be used in the output layer for the binary classification, softmax - for multiclass classification or linear - for regression type of problem [36].

For the hidden layers, one of the most popular options is the rectified linear unit

(relu), which cancels out negative inputs while keeping positive ones. Because of its simplicity, relu and its variants are computationally efficient. Furthermore, they speed up training as they avoid the vanishing gradient problem [74]. However, a problem known as "dying relu" affects relu activation functions, where neurons become inactive during training. In such cases, variants such as parametric relu or leaky relu can be implemented to mitigate this problem. Activation functions, such as tanh or sigmoid, produce zero-centred output, which can be useful in some situations, particularly when used in the hidden layers. Furthermore, they provide a smooth transition between values, which may not be ideal for optimisation. This demonstrates the value of experimenting with various activation functions during the model-development stage to observe how they impact the neural network's performance during training for the given task. Specifically, as the architecture of the neural network, the use of regularisation techniques and other hyperparameters may impact the choice of activation function.

4.2 Implementation of Complex-Valued Multilayer Perceptron

In optical communication, the task of the equaliser is to reduce the dispersive effects interfering with the nonlinearities present in the channel and/or devices [75]. Thus, the equaliser must be able to solve a regression problem where a block of consecutive symbols is used to equalise the symbol in the middle of the block. This work considers Complex-valued Neural Networks (CVNNs) as they are more effective in capturing the sequential relations between complex samples as the real and imaginary parts of complex-valued numbers are not separated. Here, two different ways of implementing complex-valued neural networks have been investigated to mitigate fibre distortions at the receiver of the optical communication link.

4.2.1 Basic implementation of Complex-valued Multilayer Perceptron

To represent the complex-valued laws describing the signal propagation, as proposed in [76; 77], the design implements complex-valued weights, activation functions, and input symbols. The figure 4.4 presents a schematic representation of the suggested NN topology.

The proposed implementation of a complex-valued multilayer perceptron is built with the TensorFlow library. This is a basic implementation of CVNNs that only uses the fundamental features of TensorFlow.

Similar to the SVM structure, the complex values of X and Y polarisations were fed into the same topology, reducing the computational complexity. Delay taps were used at the NN architecture's input to study the channel memory effect. Thus, each symbol received was equalised using a vector of consecutive samples before and after the sample of interest. The received symbols were then used to form the neural network's feature

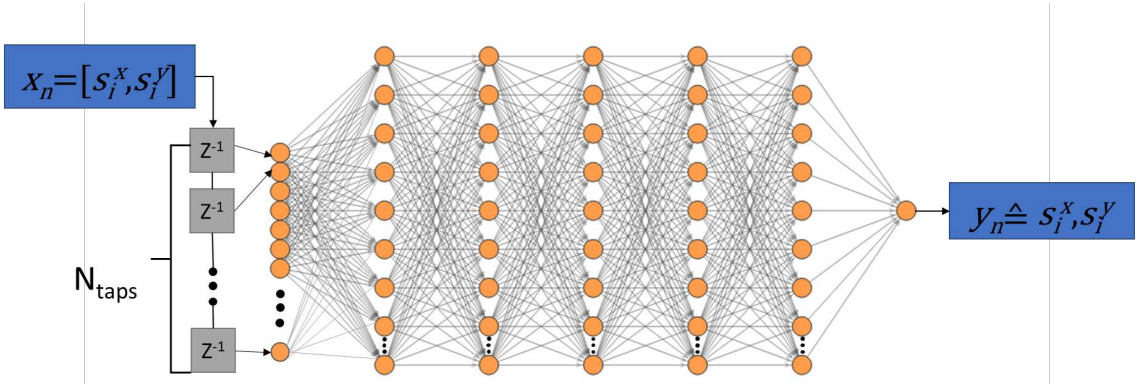

Figure 4.4: The Architecture of the Multilayer Perceptron-based equaliser, where s_i^x is the ith symbol of polarisation X, N_{taps} is the number of delay taps

vector. The size of the input layer was $2(N_{taps}+1)$, where N_{taps} is the number of delay taps.

Bayesian optimisation is used to optimise this MLP's configuration (including the number of layers and neurons in each layer) for all possible combinations of launched power and input vector sizes. In this case, all layers, except the first one, are assumed to have an equal amount of neurons, so the resulting configuration is sub-optimal. As a result, the results of such optimisation can be obtained quickly.

The activation function is an integral part of the MLP. With the exception of the final layer, where a linear function was used, all layers, in this case, used the Hyperbolic Tangent (tanh) activation function Eq. 4.1 to both the real and imaginary parts. As a result, our equaliser is a regression machine learning solution that aims to reduce the impact noise—a complex and nonlinear intermix of noise and signal—from the received QAM symbol.

$$f(x) = \tanh(\Re(x)) + j * \tanh(\Im(x)) \tag{4.1}$$

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{4.2}$$

where x is the input tensor.

A mean square error loss function (Eq. 4.3) is used for training, and an Adam optimiser with a learning rate of 10^-5 is selected for backpropagation.

$$L_{MSE} = \frac{1}{M}\sum_{i=0}^{M-1} |y_{pred} - y_{target}|^2, \tag{4.3}$$

where y_{pred} are the predicted symbols, y_{target} are the desired symbols (transmitted), and M is the size of the batches.

4.2.2 Results with the basic implementation of Complex-valued Multilayer Perceptron

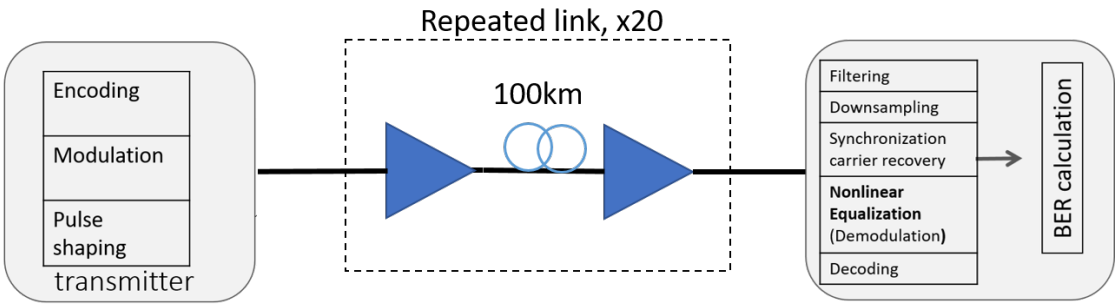

Figure 4.5: The Scheme of simulated transmission link for testing Complex-valued neural network-based equalisers

To evaluate the performance of the suggested implementation of the neural network-based equaliser, the data transmission has been simulated, as shown in Fig. 4.5. The system comprises of 32-Gbaud 16-QAM polarisation division multiplexing transmitter, 20 spans of 100 km SSMF ($\alpha = 0.2$ dB/km, $D = 17$ ps/nm/km, $\gamma = -1.3$/W/km), an erbium-doped optical amplifier with a noise factor of NF = 4.5 dB, used after each span to compensate for losses, and a coherent receiver. After the polarisation components had been separated in the receiver, the signal went through a matched root-raised-cosine (RRC) filter with a 0.1 roll-off value. Then, cumulative chromatic dispersion was corrected in the frequency domain, and the signal was downsampled to 1 sample per symbol. The training process uses 2^{18} pairs of complex-valued dual polarisation samples and the associated target transmitted QAM signal. The training, validation and test sets are divided into 70, 20, and 10 per cent of the dataset, respectively. The training set was shuffled at the beginning of training to prevent overfitting resulting from learning the correlations between adjacent symbols [78]. The NNs equaliser was trained for 500 epochs total with a batch size of 2000.

Next, the suggested NN code was used to account for nonlinear effects, after which the signal was demodulated, and the BER was calculated.

Figure 4.6 demonstrates the results of the basic implementation of complex-valued NNs. From there, it can be noticed that this NN only provides an improvement in the nonlinear regime (at values higher than optimum signal power). From there, it can be concluded that the equaliser is indeed tackling fibre nonlinearities. At the optimum launch power, the BER improvement is almost 0.17 dB. The value of optimum signal power stayed the same. Figure 4.7 demonstrates the constellations before and after the equalisation for the case of the optimum signal power. From there, a clear improvement in the separation of the clouds can be seen.

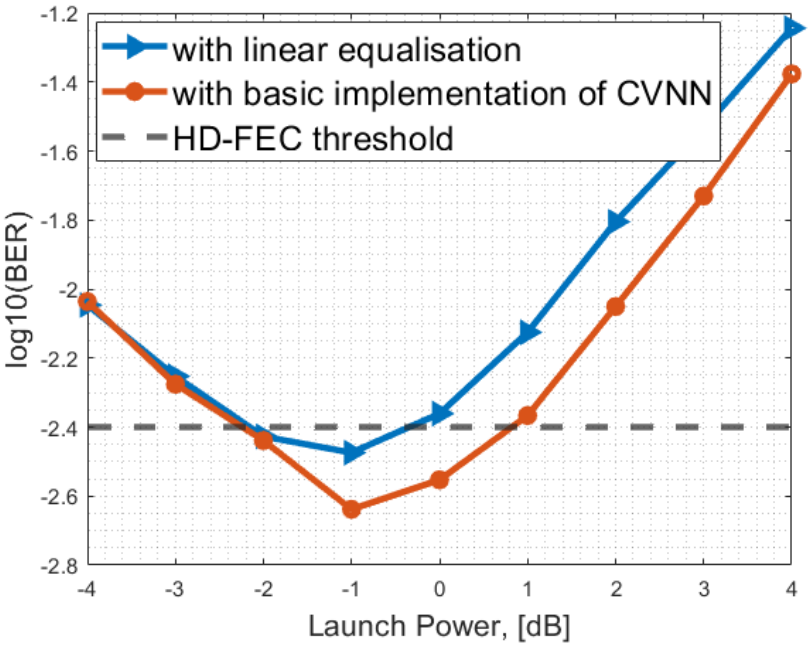

Figure 4.6: BER vs signal power with the initial implementation of Neural Network equaliser at the receiver

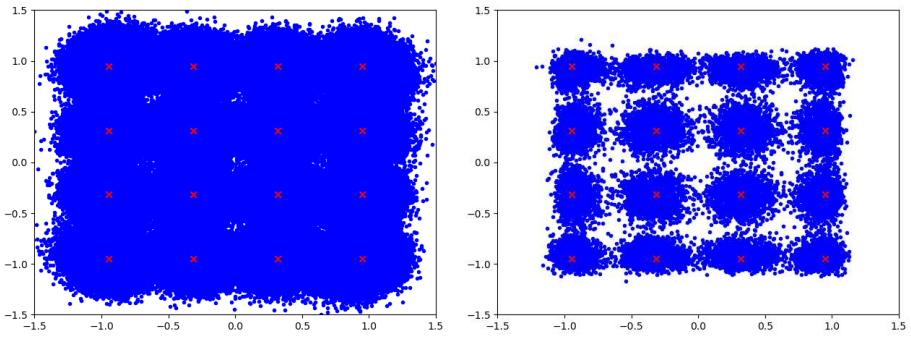

Figure 4.7: Constellations a) before and b)after equalisation with the basic implementation of NNs at -1dBm signal power for the 16QAM 2000km transmission link

4.2.3 Keras implementation of Complex-valued Multilayer Perceptron

The following section describes the implementation of Complex-valued Neural Networks using TensorFlow Keras-Functional API. This was developed utilising the additional features of the Keras and based on a library [79]. To avoid overfitting the NN model, $l2-$regularisation, early-stopping, and dropout layers have been added. Furthermore, the Custom Callback was included to calculate the change in BER between epochs. The number of neurons, layers, dropout rate, and L2 regularisation parameter were optimised using Bayesian optimisation for all combinations of launched power and input vector sizes.

The range of possible values for the number of layers was l in [2, 6], while the range of neurons evaluated was n in [50, 800]. The regularisation penalty was set at $L2$ in $[10^{-4}, 10^{-1}]$, and the dropout rate was set at p in [0, 0.5]. The input layer is made up, as before, of a series of successive samples from the two polarisations with varying sizes.

Through extensive testing, the Cartesian Rectified Linear Unit (relu) was chosen for the activation function [80] (see Eq 4.4), with the exception of the final layer. As before, 2^{18} complex-valued samples were used for training, split into 70, 20 and 10 percent for training, validation, and testing, respectively.

$$f(x) = MaxValue, \text{ for } x >= MaxValue, \tag{4.4}$$

$$f(x) = x, \text{ for } threshold <= x < MaxValue, \tag{4.5}$$

$$f(x) = \alpha(x - threshold), \text{ otherwise} \tag{4.6}$$

The values for the epochs and the batch size are the same as before, 500 and 2000, correspondingly. Weights are learned from propagating data using Adam optimiser after initialisation by He uniform [81], which minimises the complex-valued mean-squared error loss (MSE). specified as in Eq. 4.3.

4.2.4 Results with the Keras implementation of Complex-Valued Multilayer Perceptron

Similarly, the equaliser's performance based on Keras implementation of the CVNN was evaluated using the dataset from section 4.2.2. The revised NN-based equaliser's BER vs signal power plot is shown in Figure 4.8. In contrast, it is clear that this CVNN implementation improves BER in both the low and high values of signal powers. It is also visible that as the system transitions from "noisy" to "nonlinear," the BER improvement grows. At maximum power, the updated NNs improved BER by almost 0.3 dB. Additionally, it can be seen that the optimum signal power grew to 0 dBm. Figure 4.9 demonstrates the constellations obtained at the optimum power. As can be seen, this NN-based equaliser resulted in a constellation with a "jail window" pattern.

Several studies have documented the existence of such constellations following the implementation of NN-based equalisation methods. [52; 82; 83; 84; 85]. The presence of

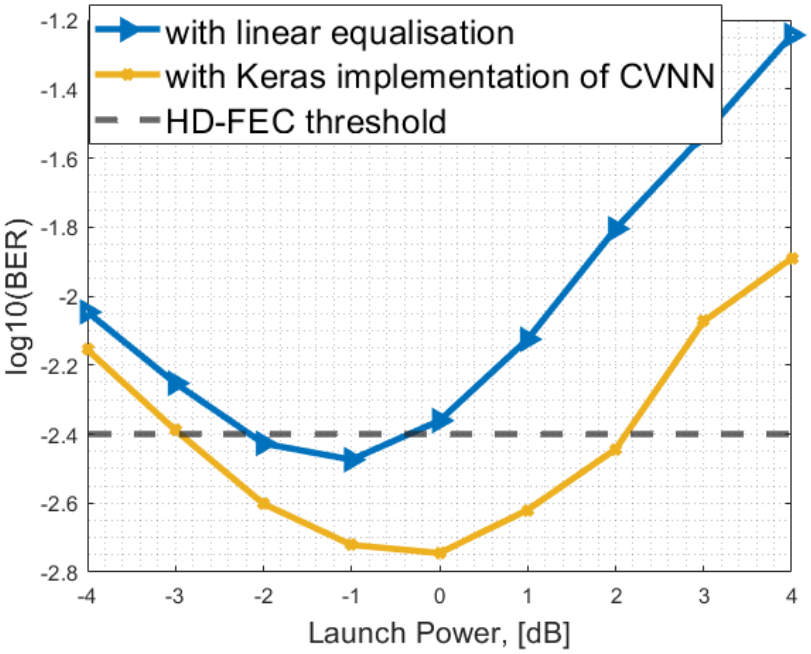

Figure 4.8: BER vs signal power with the Keras implementation of Neural Network equaliser at the receiver

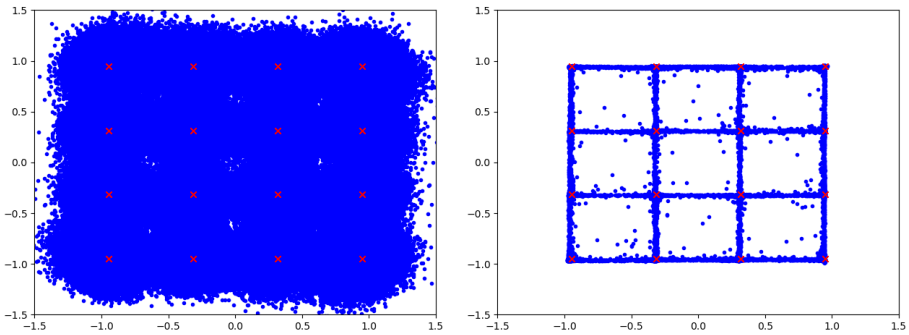

Figure 4.9: Constellations a) before and b)after equalisation with the Keras implementation of NNs at 0 dBm signal power for the 16QAM 2000km transmission link

the "jail window" phenomenon arises from the regression task performed by NN equalisers using the Mean Squared Error (MSE) loss [8]. This effect occurs because these NNs aim to minimise the Euclidean distance between the recovered and transmitted symbols. This effect results in the formation of non-Gaussian constellations, which violate the Gaussian channel assumption used in the computation of specific QoT metrics, such as EVM and MI. As a result, the accuracy of these Gaussian assumption-based metrics is reduced, and the quality of transmission may be overestimated when these inaccurate metrics are used. This phenomenon can result in the NN appearing to perform well based on these inaccurate metrics, while the true gain provided by the "jail window" constellation is overestimated. This highlights the importance of choosing the correct QoT metric to measure the resulting performance of the ML-based nonlinearity equalisation.

The occurrence of the "jail window" effect can also be attributed to the discrepancy between the actual transmission performance metric, BER, and the metric minimised during NN training, which is the MSE loss. This mismatch leads to a disagreement between the objective function and the NN's actual output. However, the BER cannot be directly utilised as a loss function for the NN due to its non-differentiable nature.

The research conducted by [8] offers valuable insights into mitigating the occurrence of "jail window" constellations and their effects on the system. Their study proposes several potential solutions, including:1)expressing the equalisation results in terms of BER or Q-factor derived from BER; 2) using larger batch size for better generalisation; 3) implementing regularisation techniques to improve generalisation.

The first and third suggestions have already been incorporated into the present implementation of neural networks. Consequently, the decision was made to focus on investigating the influence of varying batch sizes on the occurrence of "jail window" constellations. Figure 4.10 demonstrates the constellations obtained for different batch sizes after NN equalisation at the optimum launch power. From there, it can be seen that increasing batch size did not help with the problem of square constellations. Additionally, the constellations at various stages of the learning process have been plotted in Fig. 4.11. From there, it can be seen despite the resulting constellations quickly transitioning to square as the number of epochs increased, it is evident that the BER of the testing data continued to improve.

Moreover, it is important to note that multi-layer perceptrons are more prone to encountering "jail window" constellations due to their feedforward structure, which allows for the possibility of hard-coding input data to the output. However, as demonstrated above, even in the presence of "jail-window" constellations, the BER results exhibit continuing improvement. This means that despite the "jail-window constellations", the performance of the transmission system is not overestimated.

Figure 4.12 illustrates the bit error rate (BER) as a function of input vector size at the optimal power of 0 dBm. The performance of the complex-valued neural network (CVNN) equaliser can be seen to improve with an increasing number of input samples.

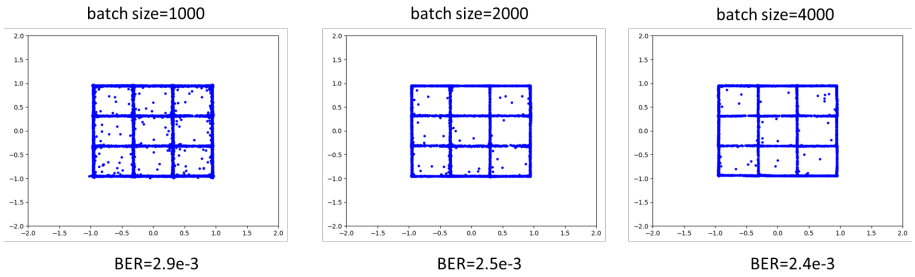

Figure 4.10: Signal constellations for different batch sizes after NNs equalisation at the optimum launch power

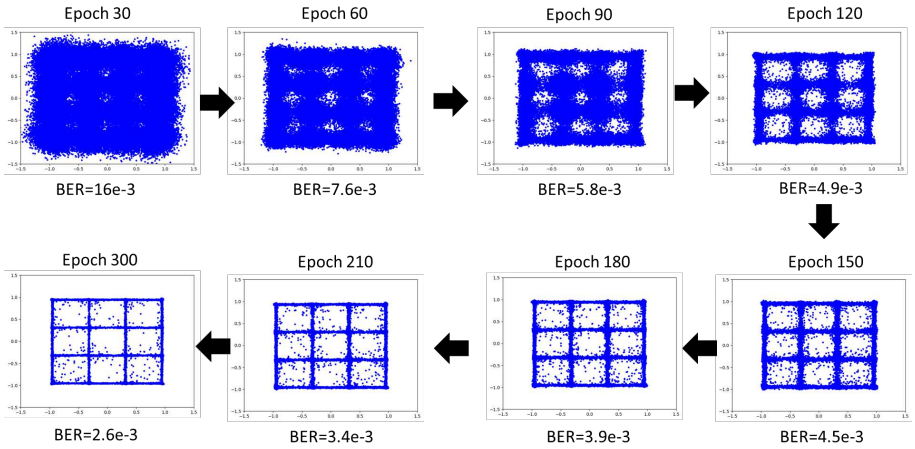

Figure 4.11: Signal constellations captured at different stages of learning

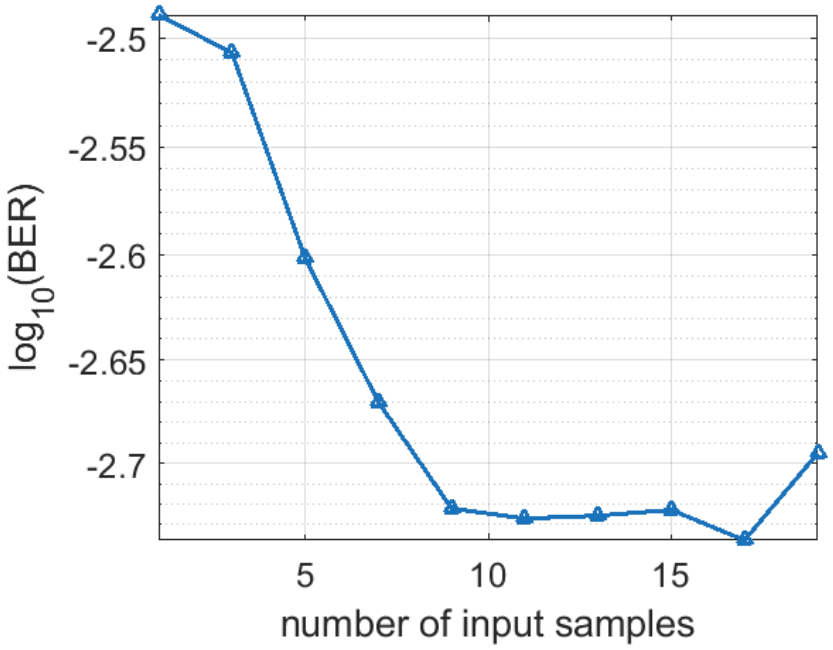

Figure 4.12: BER at optimum power with CVNN equaliser based on Keras implementation vs the input vector size.

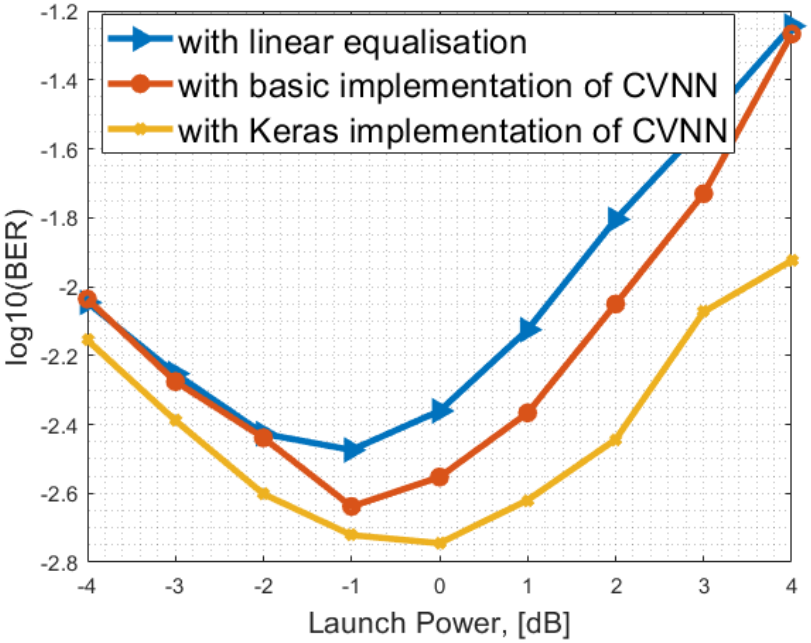

Figure 4.13: The Comparison of the results between the basic and Keras implementation of Complex-valued NNs.

Additionally, the figure demonstrates that a minimum of 10 delay samples is required to attain maximum performance.

The difference between the basic and Keras CVNN equaliser implementations is depicted in Figure 4.13. The Keras CVNN implementation is shown to outperform the basic version. From the constellation plots, it is evident that the Keras CVNN equaliser better separates the clusters, though resulting in "jail window" constellations. Therefore, only a Keras neural network-based equaliser configuration will be considered from now on.

4.3 Computational Complexity Analysis of Multilayer Perceptron

As was previously stated, computational complexity is another important metric for the application of NN-based equalisers in real optical communication systems. The computational complexity of an MLP depends on several factors, including the number of layers, the number of neurons per layer, the type of activation functions used, and the optimisation algorithm [36]. Here, the complexity is measured as the number of operations (multiplications) necessary for an MLP's inference step. Specifically, for a single-layer feedforward pass, the complexity can be expressed as $O(n_i * n_{i-1})$, where n_i represents the number of neurons in the current layer, and n_{i-1} denotes the number of neurons in the previous layer. The CC of MLP is highly dependent on the number of layers and neurons in each layer, with deeper networks and more neurons typically demanding increased computations. Additionally, both the processing delay and the computational load increase

Table 4.1: Optimised CVNN architectures for different launch powers

Launch Power	Optimal Parameters
-4 dBm	$n = [800, 50]$, $l = 2$
-3 dBm	$n = [800, 50]$, $l = 2$
-2 dBm	$n = [800, 50]$, $l = 2$
-1 dBm	$n = [800, 800, 50, 50, 800, 800]$, $l = 6$
0 dBm	$n = [800, 800, 50, 50, 800, 800]]$, $l = 6$
1 dBm	$n = [800, 800, 800, 50, 800, 800]$, $l = 6$
2 dBm	$n = [800, 800, 800, 50, 475]$, $l = 5$
3 dBm	$n = [800, 800, 50, 50, 50, 50]$, $l = 6$
4 dBm	$n = [800, 396]$, $l = 2$

with the number of input samples. The complexity is also influenced by the selection of activation functions. For example, relu proves computationally more efficient compared to sigmoid and hyperbolic tangent (tanh) due to its straightforward thresholding mechanism. Furthermore, the integration of regularisation techniques, such as dropout or weight decay, introduces an extra layer of computations during the training phase.

The computational complexity of the Multilayer Perceptron, expressed as a number of multiplications, can be calculated using the following formula:

$$CC = (n_s n_i n_1 + \sum_{l=1}^{L-1} n_l n_{l+1} + n_o n_L) \tag{4.7}$$

where n_s is the memory size (e.g. $2N+1$, being N the number of taps), ni is the number of input features, n_o is the number of outputs, and n_l corresponds to the number of neurons in each layer with $l \in [1, L]$.

It should be noted that the performance of the Complex-valued NNs was measured as the number of complex-valued multiplications. Table 4.1 shows the optimised NN architectures for a given launch power/number of delay taps. Using the formula from Eq.4.7, the graph was plotted, demonstrating the performance of the NN-based equaliser (as BER) vs the computational complexity (expressed as a number of multiplications).

In Fig. 4.14, the trade-off between performance and complexity can be observed. It shows that the optimal launch power lies where the optimised neural network (NN) equaliser reaches its peak computational complexity. Moreover, it is evident that better bit error rate (BER) performance requires a more computationally complex equaliser. Furthermore, the equaliser's complexity in the "noisy" region is considerably lower than in the "nonlinear" region. This is because, in the "noisy" region, there is a minimal improvement over linear equalisation since the NN struggles to eliminate the random noise. Consequently, a small NN equaliser is adequate to achieve the level of linear equalisation.

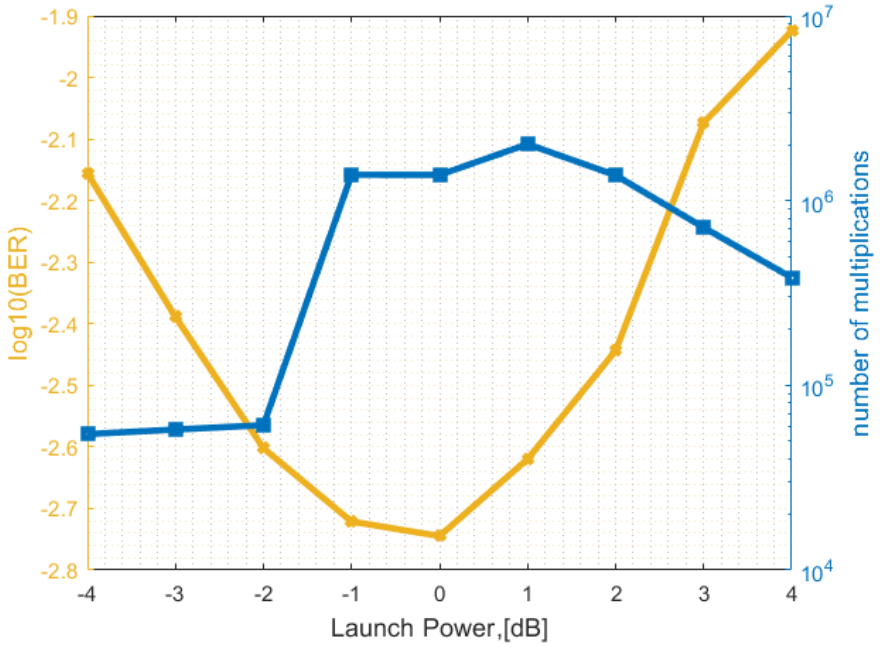

Figure 4.14: The computational complexity of Keras implementation of CVNN for each value of launch power.

4.4 Strategies for Computational Complexity Reduction

As this paper focuses on low-complexity equalisers, it is important to investigate different approaches for reducing the computational complexity of the resulting NNs. There is a growing inclination towards utilising over-parameterised architectures when designing neural networks. This is primarily done to ensure superior model performance and take advantage of the improved learning capabilities [86; 87]. The underlying reason for this preference is the impact of a larger parameter count on the smoothness of the loss function. This smoothness facilitates the convergence of gradient descent techniques [86]. However, the trade-off for such over-parameterisation is the increased need for computational and memory resources [88; 89]. As a result, considerable effort is being put into creating methods that can aid in simplifying the NNs without significantly degrading their performance, as evidenced by [88; 89; 90]. One of them is the pruning method, which eliminates redundant NN components to reduce the size and computational complexity of the network.

4.4.1 Weight Pruning

In machine learning, pruning is the process of removing components from a model, such as weights or nodes, in order to simplify, reduce complexity, and potentially improve its performance. Neural networks are frequently subjected to weight pruning, but the techniques employed can change based on the model type.

There are several benefits to implementing pruning, such as [86; 90]:

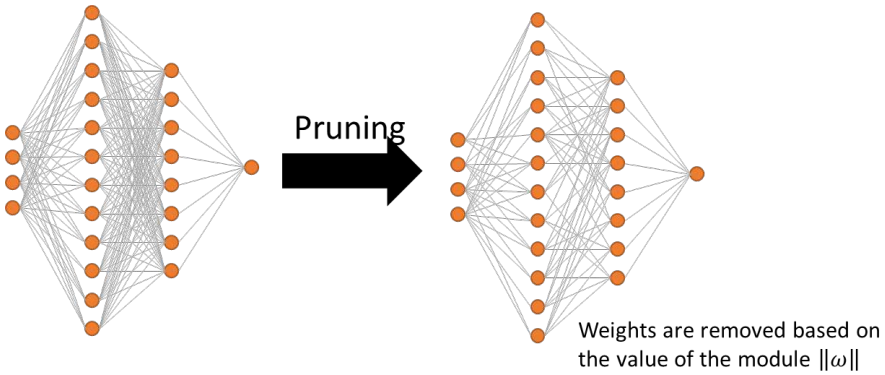

Figure 4.15: The concept behind weight pruning.

- To reduce overfitting: A complex model with too many parameters may overfit the training set, resulting in poor generalisation of new data. Pruning, which involves discarding the model's less significant components, can help reduce overfitting.
- To increase computational efficiency: Pruning can reduce the time and computational resources needed for training and testing stages by decreasing the number of parameters in a model.
- To improve memory footprint: frequently pruned models have a smaller memory footprint, which makes them better suited for deployment in environments with constrained resources.

Several techniques exist to perform pruning, including weight, neuron or tree pruning, each targeting different components of the model. Pruning can be applied to weights, biases, and activations, although there is minimal incentive to reduce biases due to their limited number and substantial impact on a layer's output. Here, the focus primarily revolves around weight pruning based on the magnitude, a process that removes low-value weights, which have minimal influence on the output, see Fig. 4.15. As the work is with complex-valued weights, and the magnitude of these weights is determined by the norm of the complex values. To determine which weights to remove, a binary criterion is applied. Weights that fulfil this criterion are given a value of zero. This ensures that the "trimmed" elements do not participate in backpropagation.

In the context of deep learning, sparsity refers to the proportion of zeros among a neural network's weights, connections, or neurons relative to its total parameters. A neural network or tensor is considered sparse when the majority of its weights or connections, or elements, respectively, are zero. Pruning aims to promote sparsity by eliminating unused connections, weights, or neurons from the network, which can produce a more accurate but compact and efficient model. The sparsity level achieved depends on the pruning technique used and the desired trade-off between model size and accuracy.

Consequently, smaller weights with less information are eliminated first, and this process continues until the desired level of sparsity (ex., % of pruned weights) is achieved.

The remaining weights are then fine-tuned through retraining to optimise the model after pruning further.

Modifying Eq. 4.7, the following equation can be used to estimate the computational complexity of the pruned model:

$$CC_{pruned} = (1-s) \cdot (n_s n_i n_1 + \sum_{l=1}^{L-1} n_l n_{l+1} + n_o n_L) \tag{4.8}$$

where s is the level of sparsity achieved by pruning the NN.

The TensorFlow Model Optimisation Toolkit — Pruning API was utilised to implement the pruning process. Notably, these frameworks are not prepared to deal with CVNN; hence, a customised implementation was carried out using the existing code.

4.4.2 Results with Pruning

This section showcases the outcomes of implementing pruning on the optimised model established in part 4.2.4 (configurations are in Table 4.1). In this analysis, various values of the sparsity of neural networks are evaluated to determine the optimal balance between reducing computational complexity and maintaining system performance (measured in BER). The subsequent figure 4.16 illustrates the consequences of applying pruning to the optimal NN model for launch powers in proximity to the optimal value. Figure 4.16 shows the performance vs. complexity trade-off that was made achievable by pruning for launch powers of 0 dBm and 1 dBm. As a result, the performance value, or BER, is shown on the y-axis. The CC values for the various NN architectures shown in Table 4.1 before and after varying levels of pruning are plotted on the x-axis using Eq. 4.8. In this instance, the pruned models' sparsity levels s varied from 20% to 90% with a 10% increment. Prior to pruning, the original models are represented by $s = 0\%$. Here, pruning is applied only to the optimised equalisers at the powers near optimum.

This demonstration aims to show that pruning can lead to simpler neural networks (NNs) while maintaining performance below the HD-FEC threshold (e.g., $BER = 3.8 \cdot 10^{-3}$). In fact, it is possible to eliminate up to 60% of the weights at a launch power of 1 dBm and still remain below the HD-FEC cutoff. With a power of 0 dBm, sparsity levels of up to 70% can be achieved while staying below the threshold of interest. Moreover, it is observed that the pruned equaliser at 0 dBm requires less computational complexity to cross the HD-FEC threshold. This corresponds to the fact that an unpruned compensation at 1 dBm demands more computational complexity than at 0 dBm.

However, even with complexity reduction through pruning, achieving a BER performance below the HD-FEC threshold with NNs necessitates processing over 10^6 complex-valued multiplications per sample. This is not feasible for real-time implementation of NN-based equalisers.

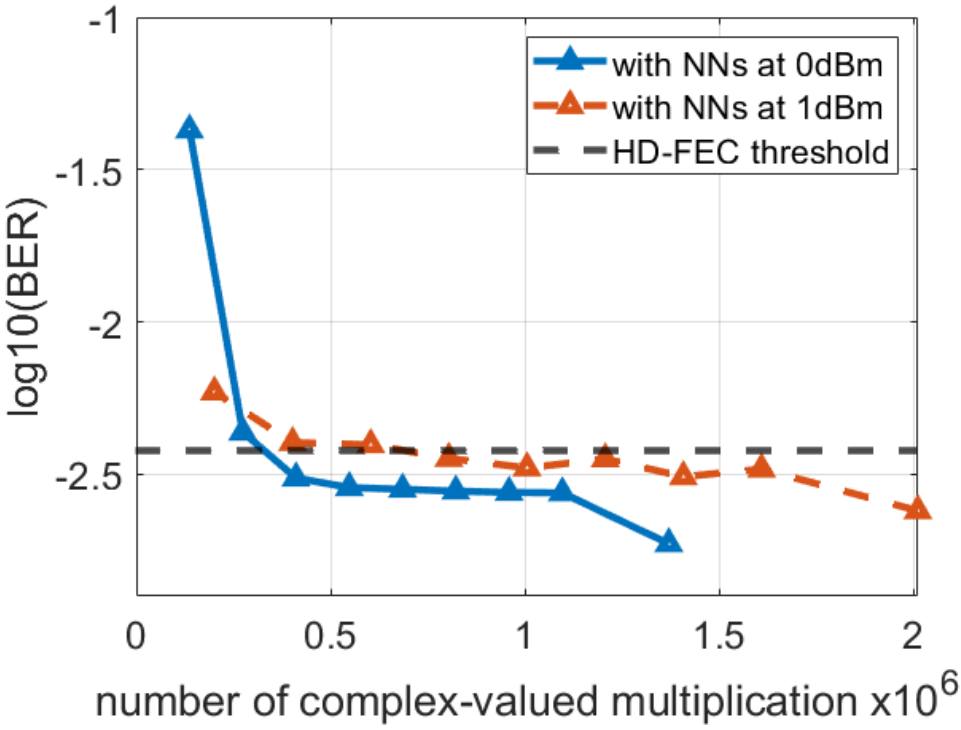

Figure 4.16: BER achieved with CVNN equalisation based on Keras for different CC values after pruning. Two launch powers are considered: 0 dBm and 1 dBm.

4.5 Conclusion

This chapter focuses on complex-valued neural network (CVNN)-based equalisation for addressing fibre nonlinearities. CVNNs offer advantages in capturing the sequential relationships between complex samples by considering the real and imaginary parts of complex-valued numbers together without separation. For this purpose, a multilayer perceptron (MLP) structure was chosen due to its low computational complexity and potential implementation using photonic devices.

Different implementations of CVNNs using various libraries were investigated, and their performance was evaluated for mitigating fibre nonlinearities in optical transmission. Both implementations demonstrated the ability to counteract the effects of fibre nonlinearities, with the improvement in BER performance increasing as the launch power rises. It should be noted that the Keras implementation of NNs resulted in a better BER performance; hence, it was selected as the preferred implementation.

To evaluate the feasibility of real-time implementation of the neural network equaliser, the computational complexity of the equaliser was assessed based on the number of complex-valued multiplications. Recognising that NNs are often over-parameterised, weight pruning was investigated to reduce the resulting computational complexity. However, even with weight pruning, the number of computations required remains high for real-time implementation of NNs.

Therefore, it is essential to explore additional methods to simplify the equaliser fur-

ther. Subsequent chapters will investigate some of the proposed techniques to reduce the resulting complexity of NNs.

Chapter 5

Combination of Neural Networks with Optical Solutions

In order to decrease the computational complexity of ML-based equalisers, an approach was considered to integrate them with optical solutions like Optical Phase Conjugation (OPC) and Dispersion Managed (DM) links. The underlying concept is that optical solutions, such as OPC and DM, can effectively address fibre distortions, reducing channel memory and minimising crosstalk between channels, particularly in scenarios involving high baud data rates and long-distance transmissions [91].

It is anticipated that employing symmetric OPC systems or completely inline dispersion-compensated links will result in a limited effective channel memory. This channel memory will have an impact on the number of required subsequent symbols for the input of the NNs, which in turn will affect the processing delay of the equaliser. However, some residual memory will always exist because of the fibre's nonlinearity and the link's asymmetry. This chapter aims to investigate the influence of OPC/DM on reducing residual memory and subsequently simplifying the Neural Network (NN) equaliser.

The subsequent section provides an explanation of OPC and DM concepts, followed by a description of their implementation. Then, the simulation/experimental setup used to test the performance of the equaliser is presented, followed by a demonstration of the results achieved using the combined equaliser. Finally, the chapter concludes with an analysis of the computational complexity of the combined approach compared to a standard NN-based equaliser.

5.1 Optical Phase Conjugation

Optical phase conjugation is a technique used in optical communication to counteract the effects of nonlinearity by reversing the phase of the light signal. This is achieved through the use of phase conjugate mirrors, which are nonlinear optical elements capable of conjugating the phase of the light signal [29]. In mid-link optical phase conjugation, a phase

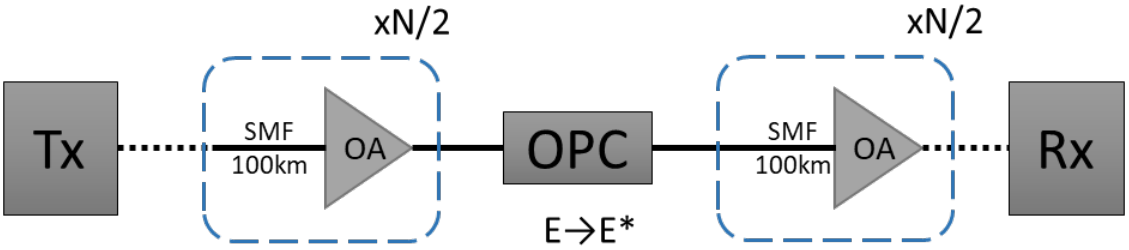

Figure 5.1: The concept behind mid-link Optical Phase Conjugation

conjugate signal is introduced in the middle of the transmission link to compensate for distortions that occur during transmission. Mid-link optical phase conjugation is typically implemented using a combination of optical amplifiers and nonlinear optical devices. The basic implementation consists of the following steps (see Fig. 5.1):

- The optical signal is transmitted along half of the overall length of the fibre link.
- In the mid-link of the link, the optical phase conjugation device applies a phase conjugation operation to the received optical signal, which generates a phase-conjugate signal. This signal is amplified to the required power level using an optical amplifier.
- The amplified phase-conjugate signal propagates across the second half of the link.
- This effectively cancels the distortions that occur during transmission.
- The compensated signal is detected and processed by the receiver.

One of the critical benefits of mid-link optical phase conjugation is that it offers a simultaneous compensation of the dispersion and nonlinearity effects of inter- and intra-channel nonlinear effects. It also allows for using higher-order modulation formats more susceptible to distortions. However, in practice, implementing mid-link optical phase conjugation can be complex and requires careful management of the optical power levels and polarisation states of the signals and the choice of the appropriate nonlinear optical device for the phase conjugation operation. Furthermore, to achieve an effective mid-link OPC, the power and dispersion profiles need to be symmetric with respect to OPC. Combining DSP and OPC techniques may make it possible to maximise their benefits while minimising their drawbacks. Examples of this strategy include relaxing the OPC design using machine learning [92] or reducing the OPC power symmetry requirement by integrating the Volterra equaliser with the OPC [93]. In this chapter, the combination of a mid-link optical phase conjugation with Neural Networks is investigated. Data for both links, regular and link with OPC inserted in the centre, were obtained from the lab [94] and used to evaluate the effectiveness of the combined equaliser approach (details below). This data was then processed by offline DSP, consisting of a multilayer perceptron.

5.1.1 Experimental Setup

The experimental configuration, which includes a polarisation division multiplexing (PDM) 28-Gbaud 64QAM transmitter, four spans of SSMF (α= 0.2 dB/km, D = 17 ps/nm/km, and γ=1.3 /W/km), and a coherent receiver, is shown in Fig. 5.2. A 45 Gbaud IQ modulator's signal source is a Continuous-wave (CW) laser set to 1555.75 nm. Four channels of AWG control the IQ modulator, which generates symbols at a 28-Gbaud baud rate. The symbols then were time-multiplexed with 5% quadrature phase-shift keying pilot symbols periodically. The symbols were oversampled and loaded in the AWG (sampling rate 56 GSa/s). At the start of each span, an EDFA (6-dB noise figure) was introduced to adjust the power. In the middle of the link, either the signal:

- passed through the OPC and amplified by an EDFA with fixed output power (15 dBm), with the conjugate of the signal propagated in the second half of the link;
- bypassed the OPC device, with a variable optical attenuator with a WSS used to simulate the insertion penalty of the OPC, where the OSNR was made to be the same in the second half of the link for the two cases (with and without the OPC).

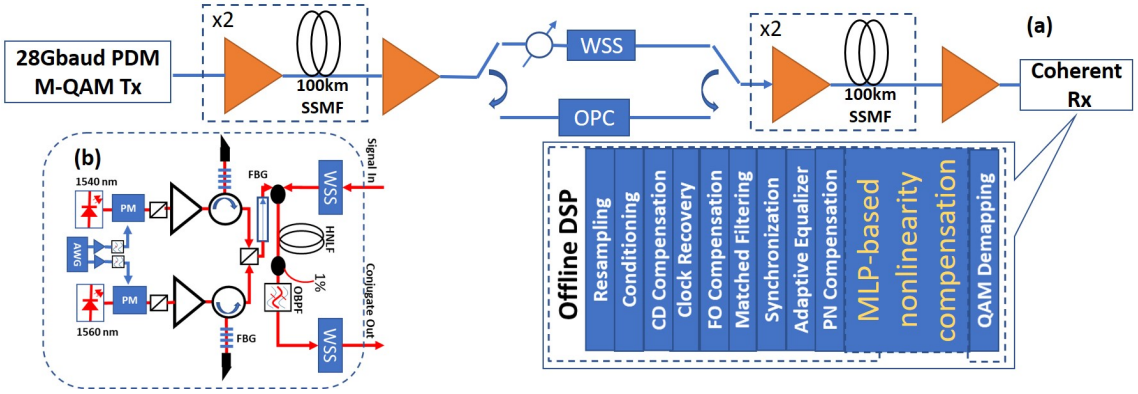

Figure 5.2: Block diagram of the experimental setup used for mid-link OPC experiments and the position of the MLP equaliser[1]

The conjugate is produced using a dual pump polarisation independent OPC [94]. Two orthogonal pumps at 1540.5 nm and 1560.1 nm (linewidths of 10 kHz and 100 kHz, respectively) were generated using counter-dithered (using two RF tones at 60 and 600 MHz) CW lasers and high-power EDFAs. The signal was then filtered and combined with the signal in 100 m of a highly nonlinear fibre (HNLF) (zero-dispersion wavelength = 1550 nm, α=1.2 dB/km, γ=21.4 /W/km, and dispersion slope = 0.041 ps/nm^2/km). The pumps are suppressed, and the conjugate is extracted using an optical band-pass filter with WSS. The local oscillator (100-kHz linewidth), whose signal was combined with the received signal in a 90° optical hybrid, makes up the coherent receiver. Four balanced photo-diodes were connected to the hybrid outputs, and a real-time sampling scope was used as an analogue-to-digital converter (100-GS/s sampling rate, 33-GHz 3-dB bandwidth). On a

desktop computer, digital signal processing with offline data was carried out, and machine learning was used before symbols de-mapping [1].

Figure 4.4 demonstrates the structure of an MLP-based equaliser used in work. Similarly, as before, real and imaginary values of X and Y polarisations were fed into the same topology, reducing the computational complexity. Delay taps were used at the NN architecture's input to study the channel memory effect. Thus, each symbol received was equalised using a vector of consecutive samples before and after the sample of interest. The received symbols were then split into real and imaginary components, forming the neural network's feature vector. The size of the input layer was $2(N_{taps}+1)$, where N_{taps} is the number of delay taps. Bayesian optimisation is used to optimise this MLP's configuration (including the number of layers and neurons in each layer) for all possible combinations of launched power and input vector sizes. In this case, all layers are assumed to have an equal amount of neurons, so the resulting configuration is sub-optimal. The activation function is an integral part of the MLP. With the exception of the final layer, where a linear function was used, all layers, in this case, used the *tanh* function. As a result, our equaliser is a regression machine learning solution that aims to reduce the impact of noise—a complex and nonlinear intermix of noise and signal—from the received QAM symbol. The training process uses 2^{18} pairs of complex-valued dual polarisation samples and the associated target transmitted QAM signal. There are 500 epochs total with a batch size of 2000. The training, validation and test sets are divided into 70, 20, and 10 per cent of the dataset, respectively. A mean square error loss function is used for training, and an Adam optimiser with a learning rate of 0.001 is selected for backpropagation.

5.1.2 Results with the basic implementation of CVNNs

The following section demonstrates the results of applying the basic implementation of an NN-based equaliser to the experimentally obtained data with and without mid-link OPC. At each power and input vector length, the optimum MLP has been applied on the PDM 28 Gbaud 64QAM signal samples with and without OPC. The results are plotted with square markers in Fig. 5.3, and they represent the BER with and without OPC and no nonlinear compensation. Figure 5.3 also shows the improvement in the performance obtained by using an MLP with 7 and 15 sample input vectors, demonstrating a significant decrease in the final BER of up to 12-fold for the case of OPC-aided MLP equaliser. As can be seen from the received constellation of the chosen points in the side panels of Fig. 5.3, this increase is made possible by a noise reduction mechanism. This improvement also results in a 4 dB rise in the optimal launch power (slightly lower than the 7% overhead HD-FEC BER) when an OPC-aided MLP-based nonlinear compensation scheme is in place, as is clear from Fig. 5.3.

At higher signal powers, the nonlinear effects become dominant, and the improvement by using the MLP equaliser also grows, see Fig. 5.4. The improvement ratio of the BER

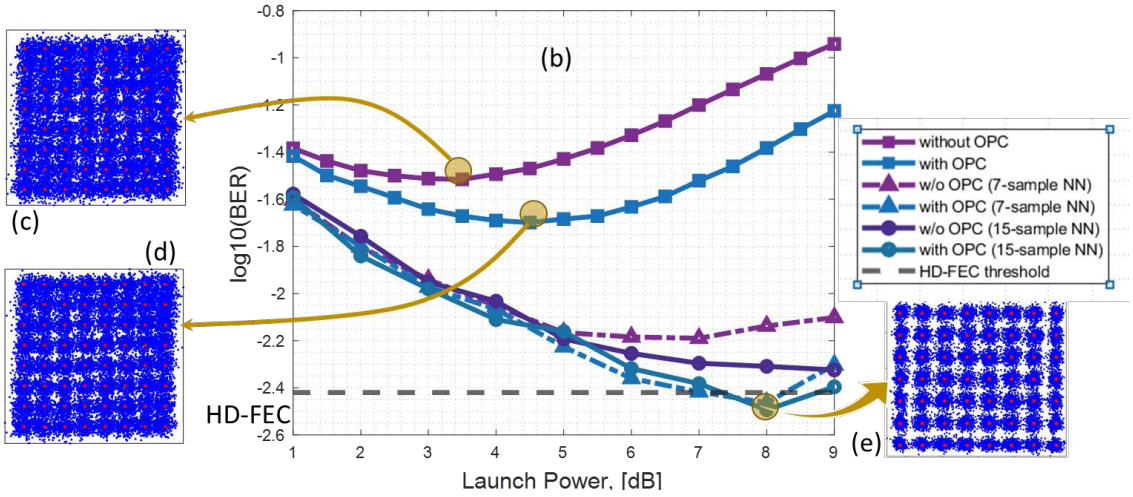

Figure 5.3: BER versus signal power and the equalised 64QAM constellation w/ and w/o OPC/NN equaliser.

for cases with and without OPC for the 7- and 15-sample input vector to the MLP is shown in this graph. This figure leads us to the conclusion that the equaliser is actually tackling the communication link's nonlinearity effects. The effect of input vector size on MLP equaliser performance is another interesting observation.

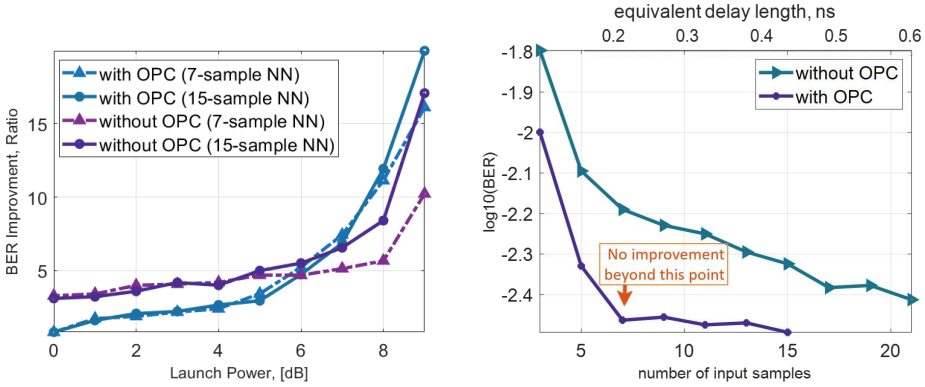

Figure 5.4: a) BER improvement w/ and w/o OPC with input vector length of 7, 15, b) BER at optimum power with NN equaliser vs the input vector size for two cases with and without OPC.

Figure 5.4b shows the BER as a function of input vector size at the optimal power of 8 dBm (for the case of OPC, but not far from the optimum power for the no OPC case as well). The equivalent delay represents the amount of time the MLP will wait to gather sufficient samples to equalise the target sample during the inference step. The size was adjusted from three samples, which are equal to the 0.1 ns channel memory (shown on top), to as much as 21 samples (equivalent to 0.6 ns). Figure 5.4b also demonstrates that in contrast to the system without OPC, the performance of the OPC-assisted system saturates beyond 7 sample input vector size. This illustrates the OPC's expected impact

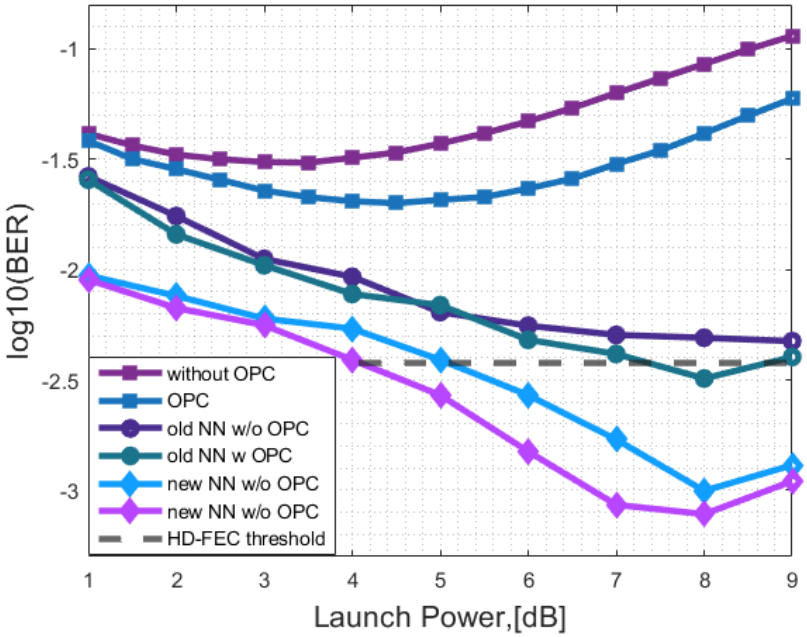

Figure 5.5: Updated BER versus signal power w/ and w/o OPC/NN equaliser.

on channel memory reduction. This memory reduction, along with channel containment, has the potential to greatly reduce the complexity of nonlinear noise reduction, especially in WDM and high bandwidth transmissions. As seen from Fig. 5.4b, even in the OPC-aided system, the MLP equaliser requires knowledge of the 3 samples that come before and after the sample of interest in order to return it to the symbol that was originally transmitted. The asymmetry of the link and defects in the devices and equipment may serve to explain the OPC system's non-zero effective memory. This link asymmetry could be reduced by using an improved symmetric link, for example, with dual pump Raman amplification rather than lumped amplification.

5.1.3 Results with the Keras implementation of CVNNs

Using the updated Complex Valued-Neural Network, described in section 4.2.3, the simulations have been repeated on the same datasets. The new Neural Network has been optimised across all taps for each value of the launch power. Also, in the structure, the regularisation parameter, early-stopping, and dropout layer were added to avoid overfitting. The new results are demonstrated in Fig 5.5, showing of comparison against the previously used NN structure. From there, it can be seen that the new structure provided a bigger BER improvement for both cases, with OPC and without, versus the previously used one. The new BER improvement for OPC aided system with an NN-based equaliser is 25 times better than BER at optimum power for the system with OPC.

Table 5.1: Comparison of the optimised CVNNs architectures for a system with no OPC and with OPC

Optical Solutions	Optimal Parameters	Power
OPC	$n = [519, 505, 531, 151, 562, 140]$, $l = 6$, $p = 0.1$, $L2 = 6.5 \cdot 10^{-6}$	8 dBm
No OPC	$n = [366, 422, 600, 600, 242, 327]$, $l = 6$, $p = 0$, $L2 = 1 \cdot 10^{-6}$	8 dBm
OPC	$n = [430, 532, 406, 462, 315]$, $l = 5$, $p = 0.35$, $L2 = 6.5 \cdot 10^{-3}$	9 dBm
No OPC	$n = [600, 600, 600, 600, 568, 50]$, $l = 6$, $p = 0.1$, $L2 = 0.1$	9 dBm

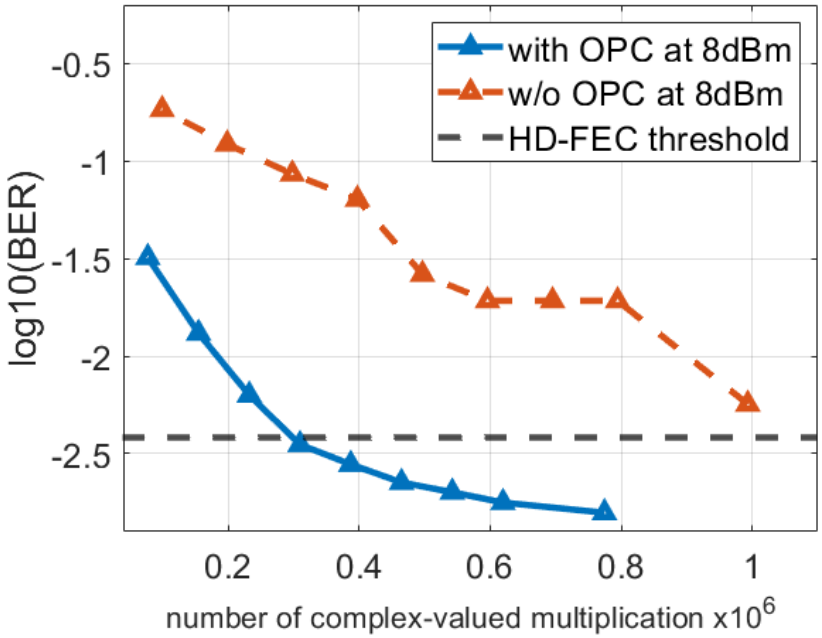

Figure 5.6: BER achieved for different CC values for two datasets where optical solutions -OPC- are used and two where were not.

5.1.4 Computational Complexity Analysis of OPC Systems

Table 5.1 contains the optimal configuration parameters for the CVNNs generated by the BO. These findings suggest that employing OPC allows for less dense layers and even fewer layers overall in terms of architecture. Similarly to before (in section 4.3), the computational complexity was calculated in terms of a number of complex-valued multiplications for the launch power near optimum using equation 4.8. Figure 5.6 displays the trade-off between performance and complexity achieved through pruning and the decision to use or not OPC at launch powers of 8 dBm with 9 taps. The y-axis represents the performance, defined as the BER, while the x-axis shows the computational complexity achieved for various NN architectures described in Table 5.1 before and after pruning to different extents. In this instance, the sparsity level s varies from 20% to 90% in 10% increments for the pruned models, while $s = 0\%$ corresponds to the unpruned original models.

Figure 5.6 demonstrates that employing OPC not only results in simpler NNs before pruning but also enables reaching higher pruning levels without exceeding the HD-FEC threshold (e.g., $BER = 3.8 \cdot 10^{-3}$). In fact, for a launch power of 8 dBm, pruning up to

60% of the weights is possible while remaining below the HD-FEC threshold. When OPC is not utilised, the optimised CVNN cannot fall below the HD-FEC threshold for any CC value at a launch power of 8 dBm. This highlights the powerful influence of OPC on simplifying NN-based equalisers' architecture. OPC not only enhances the performance of the optical communication system (see [1]) but also aids the pruning technique in reducing the equaliser's CC.

5.2 Dispersion Managed Links

Previous research has shown the benefits of using machine learning to mitigate nonlinearities in dispersion-unmanaged optical communication systems. However, a dispersion-managed link will affect signal propagation differently than a dispersion-unmanaged link. In uncompensated transmission lines, the accumulation of dispersion causes significant temporal spreading of the signal. This results in an increase in the channel memory, which is directly proportional to the length of the link. Additionally, the combination of large channel memory and nonlinear transmission effects leads to the emergence of long-memory nonlinear distortions. As a consequence, processing these distortions requires more time and computational power, especially at higher symbol rates. To mitigate these effects, dispersion management techniques can be employed to reduce the effective channel memory. However, compared to uncompensated links, dispersion management typically necessitates the use of additional optical amplifiers, which can introduce amplified spontaneous emission (ASE) noise and decrease the signal-to-noise ratio. It may also amplify certain nonlinear effects in the fibre. Various tools have been explored to address the challenges posed by nonlinearity in optical links, among which NN equalisers have shown effectiveness. By utilising NN equalisers, the detrimental impact of nonlinearity can be reduced. Therefore, for certain applications, employing NN equalisers might offer a beneficial trade-off between managing channel memory and mitigating nonlinear effects.

The potential of a DM system to simplify the computational complexity (CC) of equalisers (such as digital backpropagation (DBP)) has been previously investigated [95]. However, since NN equalisers have shown impressive performance in mitigating nonlinearities, it is essential to study the reduced-complexity version of these NN-based equalisers in the context of DM links. This can balance performance and complexity, making it a more practical solution for real-world optical communication systems, especially for future real-time implementations.

As discussed earlier in chapter 2, the effects of chromatic dispersion in the optical communication system can be effectively compensated using linear digital signal processing techniques. Digital dispersion compensation allows for reducing the number of optical amplifiers due to removing additional signal attenuation in optical dispersion compensating elements, substantially decreasing accumulated optical noise of the link and, thus, improving overall system performance. The other way of compensating for the dispersion

effects is to apply an all-optical solution, one of which is to use Dispersion Compensated Fibres (DCF). DCF is designed with a high negative dispersion coefficient that cancels out the positive dispersion of standard single-mode fibre (SMF), which is commonly used in long-haul optical communication systems. DCF can be implemented in different forms, such as a separate fibre segment or a module containing a DCF fibre of a short length (Dispersion Compensating Module or DCM). DCFs are designed with a carefully engineered refractive index profile and are typically made of silica or fluoride glass. Depending on the specific requirements of the communication system, DCFs can be designed with either positive or negative dispersion. When used in conjunction with optical fibre, the DCF helps counteract chromatic dispersion's effects and reduce signal distortions. With the inclusion of the DCF fibres, the Nonlinear Schrodinger equation from Eq. 2.1 can be rewritten as:

$$i\frac{\partial A}{\partial z} - \frac{1}{2}\beta_2^{(1,2)}\frac{\partial^2 A}{\partial t^2} + \gamma^{(1,2)}|A|^2 A = \\ = i[-\alpha^{(1,2)} + \sum_{k=1}^{N} {r_k}^{(1,2)}\delta(z - z_k^{(1,2)})] = iG^{(1,2)}(z)A \quad (5.1)$$

where subscripts 1 and 2 correspond to SMF and DCF, and $z_k (k = 1, ..., N)$ are locations of amplifiers and ${r_k}^{(1,2)} = [exp(\alpha^{(1,2)} z_a^{(1,2)}) - 1]$ are amplification coefficients for SMF and DCF parts.

The dispersion-managed optical communication system is the system that includes such fibres periodically along the system length so that the optical pulses reaching the receiving end will have zero or near-zero dispersion [96]. Figure 5.7 demonstrates the dispersion parameter D values across the link. From this figure, it can be seen that for this configuration, the mean of the dispersion is almost zero.

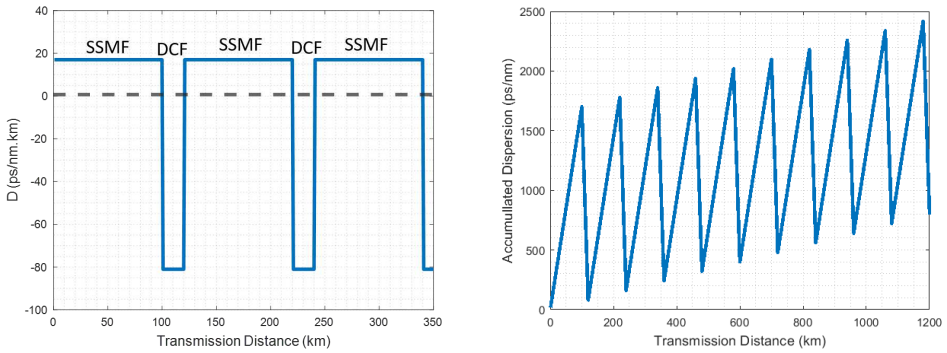

Figure 5.7: a)Variation of the dispersion parameter D and b)Accumulated dispersion across the dispersion-managed link

Various dispersion management configurations involve the strategic placement of optical components. Pre-compensation techniques are used in dispersion management to control dispersion before it happens. These techniques are often used at the optical trans-

mitter end because dispersion occurs within optical fibres during signal transmission [97]. On the other hand, post-compensation dispersion management techniques are used to manage dispersion after it happens and are typically conducted at the optical receiver end. The goal is to counteract the phase factor that causes dispersion-induced degradation of the optical signal during fibre propagation [9]. For the inline dispersion compensation, dispersion compensating elements, such as Dispersion Compensating module (DCM) or Dispersion compensating fibre (DCF), are periodically integrated along the system length and often positioned at amplifier sites. Therefore, in the context of inline pre-compensation or inline post-compensation schemes, the implementation involves introducing a periodic alternation of SMF and DCF fibres in an appropriate order. Optical amplifiers are strategically placed between them to mitigate fibre loss. Another example of inline dispersion compensation configuration is a symmetrical compensation, which involves balancing pre-compensation and post-compensation techniques to effectively address dispersion-related issues [98].

In dispersion-unmanaged links (DUM), the cumulative chromatic dispersion increases linearly with distance. On the contrary, in dispersion-managed (DM) systems, the dispersion introduced by the transmission fibre is partially mitigated at the end of each span using dispersion compensating fibre. There is typically a residual dispersion remaining at the end of the link. Figure 5.7 demonstrates the cumulative dispersion D as a function of the length of the transmission link. The bandwidth of such a communication system is much wider to the extent that it can remove dispersion from all channels simultaneously, and the power consumption can be smaller. Since inline dispersion-managed systems can be used to compensate for fibre dispersion fully, they can simplify the digital signal processing at the receiver. However, one of the drawbacks is the enhancement of nonlinear effects due to the small mode diameter of DCF fibres for high input optical powers. In addition, inline dispersion-managed systems lead to an increase in power loss due to fibre attenuation in DCF and insertion loss. The amplifier gain must be raised to compensate for this loss, which raises the ASE noise[97]. Overall, the dispersion-managed link provides a trade-off: it increases the nonlinearity effects of the link, but due to inline dispersion compensation, it decreases signal distortions.

In the next part, the potential of inline dispersion management to simplify the NN equaliser's architecture while maintaining comparable performance to the uncompensated link (also using NN-based equalisation) is investigated. To achieve this, various configurations of inline dispersion compensation have been examined.

5.2.1 Results

Depending on the placement of SMF, DCF and amplifiers, various possible configurations exist for inline dispersion-managed links. Here, four periodic links have been considered built of the basic cells depicted in Fig. 5.8. The SD building block corresponds to the

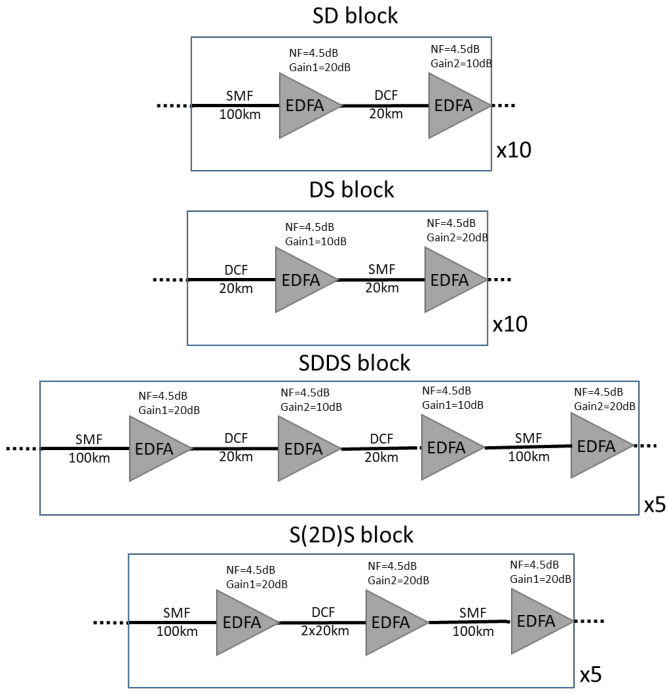

Figure 5.8: Composition of four different elementary blocks of the considered periodic DM links

.

span of SMF followed by an erbium-doped fibre amplifier (EDFA) compensating for SMF loss, then the span of DCF followed by an EDFA compensating for DCF loss. This can be thought of as a basic inline "post-compensation" cell. The block DS starts with the span of DCF, EDFA to mitigate DCF loss, and then SMF and EDFA to recover signal power after SMF, which can be considered an inline "pre-compensation" cell. The SDDS block shown in Fig. 5.8 comprises of SD cells followed by DS cells, corresponding to the "symmetric compensation". In this case, the number of links would be half of the SD and DS blocks. Additionally, a new scheme, $S[2D]S$, is suggested, which corresponds to an elementary cell starting from 100 km of SMF followed by the dispersion-compensating module that includes EDFA compensating for SMF loss, then a span of 40 km DCF and EDFA compensation for DCF loss and then again 100 km SMF and EDFA mitigating SMF span loss.

Figure 5.9 demonstrates the simulation setup used in this section, with variations of the inline dispersion management cell shown in Fig. 5.8. A single channel dual polarised transmission at 32 GBaud using a 16 QAM signal has been considered with a sampling rate of 4 samples/symbol. The dispersion managed link consists of 10 spans of 100 km SSMF and 20 km DCF ($\alpha = 0.2$ dB/km,$\alpha_{dcf} = 0.5$ dB/km, $D = 17$ ps/nm/km, $D_{dcf} = 85$ ps/nm/km, $\gamma = -1.3$/W/km, $\gamma_{dcf} = 2.8$/W/km) in various configurations. The dispersion compensating module (DCM) is believed to contain the DCF fibre in these instances, and its length is not considered when determining the transmission's overall length. Each of these links (except $S[2D]S$) contains 20 amplifiers, which are placed between each span of

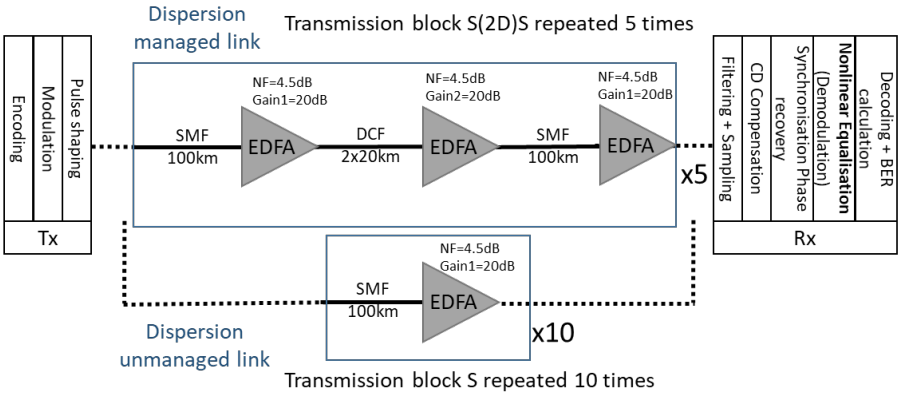

Figure 5.9: The configurations of the dispersion-managed and dispersion-uncompensated links.

SMF and DCM. At the receiver, the signal is filtered using a raised-cosine filter with a roll-off factor of 0.001, down-sampled (2 symbols/sample), and then subjected to conventional DSP to compensate for the linear effects, including residual dispersion. The results of the inline dispersion-managed links are compared to an unmanaged link consisting of 10 spans of 100 km SSMF with ten amplifiers between spans and full chromatic dispersion compensation at the receiver.

For the training of the Neural Networks, 2^{18} input samples were generated at each value of launch power for each of the DM link configurations. As before, 70% of the data was used for training the model, 20% for validation and 10% for testing. Bayesian optimisation was used to determine the number of layers, neurons, regularisation parameters, and other hyperparameters of the neural network. The performance of the neural network-based equaliser was evaluated by calculating the BER for the testing dataset.

The BER versus launch power plots for each link setup, when only linear equalisation was used, are shown in Fig. 5.10. The plots show that the dispersion unmanaged link outperforms the managed link when only linear equalisation is applied. This is because the dispersion-managed link suffers from 1) increased nonlinearity due to the higher nonlinearity of DCF fibres and 2) added ASE noise, as it requires twice as many amplifiers as the unmanaged link. As also can be seen, none of the link configurations produced BER values below the HD-FEC threshold without the use of an NN-based equaliser.

The BER improvement achieved by applying an NN-based equaliser at the receiver is demonstrated in Fig. 5.11 for all link configurations. Compared to the results of linear compensation, the optimum launch power has increased, especially in the links containing DCF fibre. This proves that the NN equaliser is effective at dealing with the excess nonlinearity caused by the use of Dispersion Compensating Fibre. The plots also show that even with machine learning at the receiver, link configurations of SD/DC can hardly cross the HD-FEC threshold. On the contrary, NN equalisation of the $SDDS$ configuration almost achieves the BER performance of the dispersion unmanaged link in addition to crossing the threshold. However, the DM link with the NN-based equaliser did not outperform,

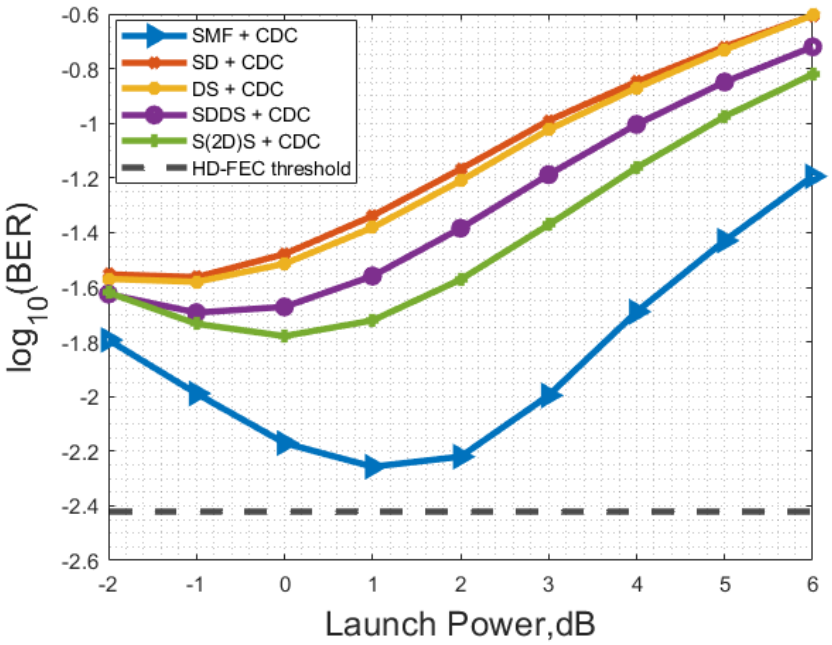

Figure 5.10: BER vs signal power for different configurations of transmission link with only linear compensation at the receiver

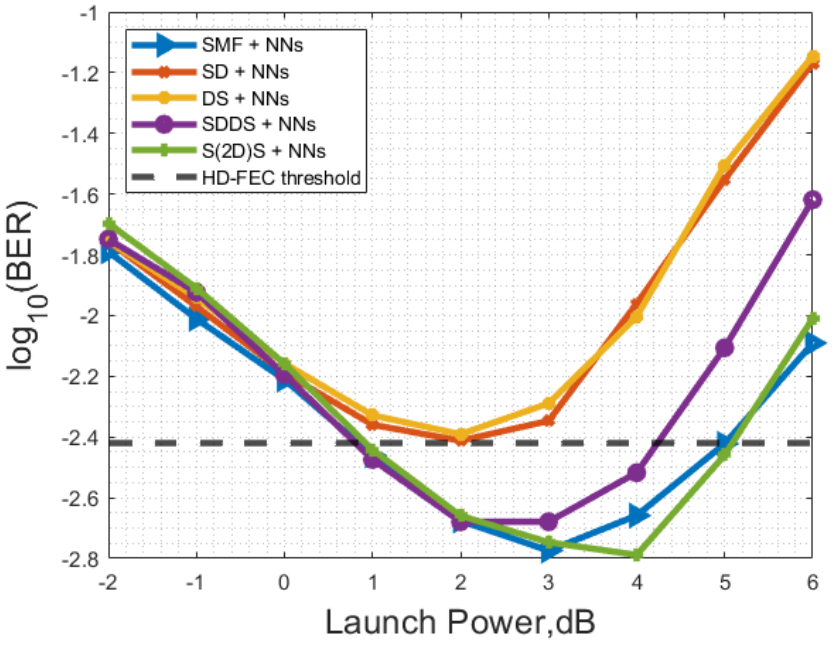

Figure 5.11: BER vs signal power for different configurations of transmission link with Neural Network equaliser at the receiver

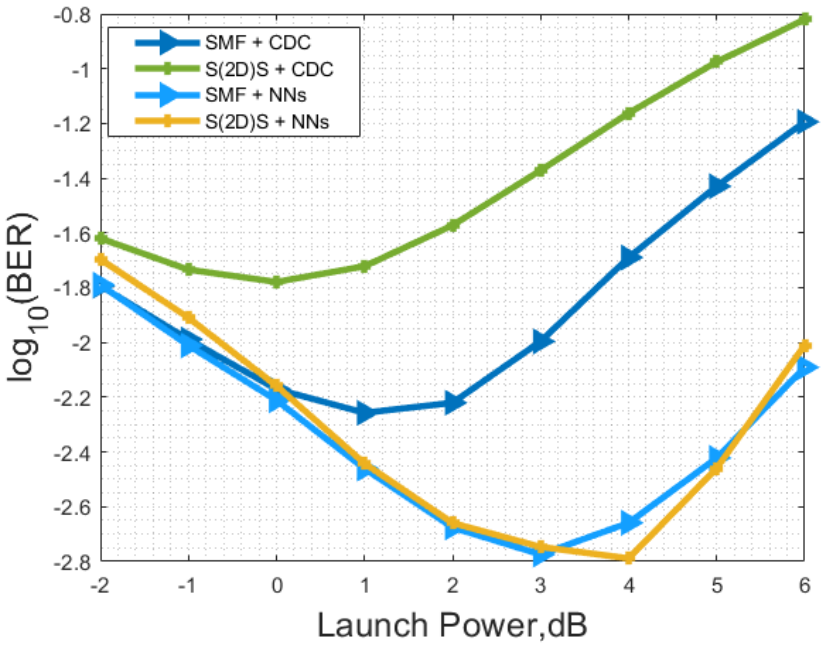

Figure 5.12: The Comparison of the results between dispersion-managed and unmanaged links with linear and NN equalisers.

most likely due to the increased ASE noise caused by the double the number of amplifiers applied compared to the unmanaged link.

An alternative structure of the $SDDS$ link, referred as $S[2D]S$ here, has been considered, which requires fewer amplifiers than the standard $SDDS$ link (15 instead of 20). Figure 5.8 (bottom box) and 5.9 show how this structure has been changed. The new configuration involves splicing together two DCF fibre spans (DCM modules) after removing the amplifier from the space in between. The BER performance of the equalisers for this link in comparison to the other link configurations can be seen in figures 5.10, 5.11. For the linear equalisation case, the performance of the $S[2D]S$ link is better than that of any other dispersion-managed links, most likely due to the fewer amplifiers, which results in the link being less affected by the ASE noise. However, the performance is still no better than using standard single-mode fibre with full chromatic compensation at the receiver. When the neural network-based equaliser is applied, the BER performance of the $S[2D]S$ link has approached the performance of the dispersion-unmanaged link. Figure 5.12 demonstrates a comparison of the results of the equalisation of these two links.

Figure 5.12 reveals that the optimum launch power rises to 4 dBm when applying neural network (NN)-based equalisation to the dispersion-managed system. Moreover, the bit error rate (BER) performance of the NN-based equaliser with the updated structure is able to attain the same level as the NNs for the unmanaged system, albeit at different power values.

The effectiveness of neural networks (NNs) was evaluated based on the input feature vector length to better comprehend the impact of inline dispersion management on channel

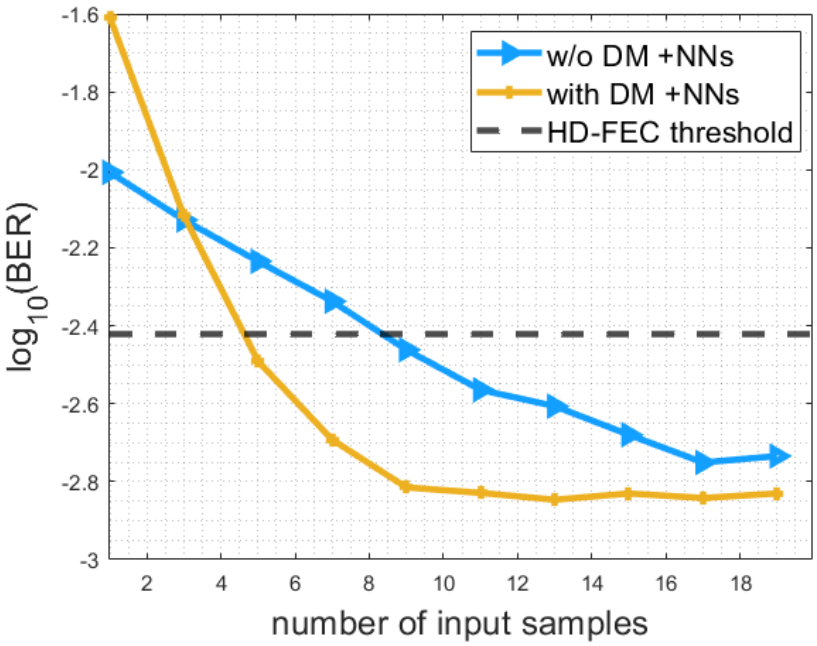

Figure 5.13: BER at optimum power with NN equaliser vs the size of the input vector for cases of dispersion-managed and unmanaged links

Table 5.2: Comparison of the optimised CVNNs architectures for dispersion unmanaged and managed systems

Optical Solutions	Optimal Parameters	Power
$S[2D]S$ link	$n = [800, 800, 50, 50, 800, 50, 50, 50]$, $l = 8$, taps=5,	3 dBm
Unmanaged link	$n = [800, 770, 50, 50, 800, 594, 800, 50]$, $l = 8$, taps=9,	3 dBm
$S[2D]S$ link	$n = [800, 50, 91, 800, 426, 800, 800]$, $l = 7$,taps=5	4 dBm
Unmanaged link	$n = [800, 678, 505, 666, 800, 800, 800, 800]$, $l = 8$, taps=9	4 dBm

memory and the potential for computational complexity (CC) reduction in the resulting equaliser. Figure 5.13 presents the bit error rate (BER) at the optimum launch power (approximately 3 dBm) as a function of the input feature vector. The figure indicates that as the number of samples increases, the NNs' performance for the unmanaged link consistently improves. In contrast, the NNs' performance for the dispersion-managed link plateaus around a certain sample input vector length. However, the dispersion-managed system achieves the optimal BER more rapidly (with a smaller input vector size). This reconfirms the channel memory reduction effect due to inline dispersion compensation provided by optical solutions. To further demonstrate that this effect contributes to the CC reduction of the resulting equaliser, a CC analysis has been conducted.

5.2.2 Computational Complexity Analysis of Dispersion Managed systems

The table 5.2 demonstrates the optimum configurations of the NNs achieved with Bayesian optimisation.

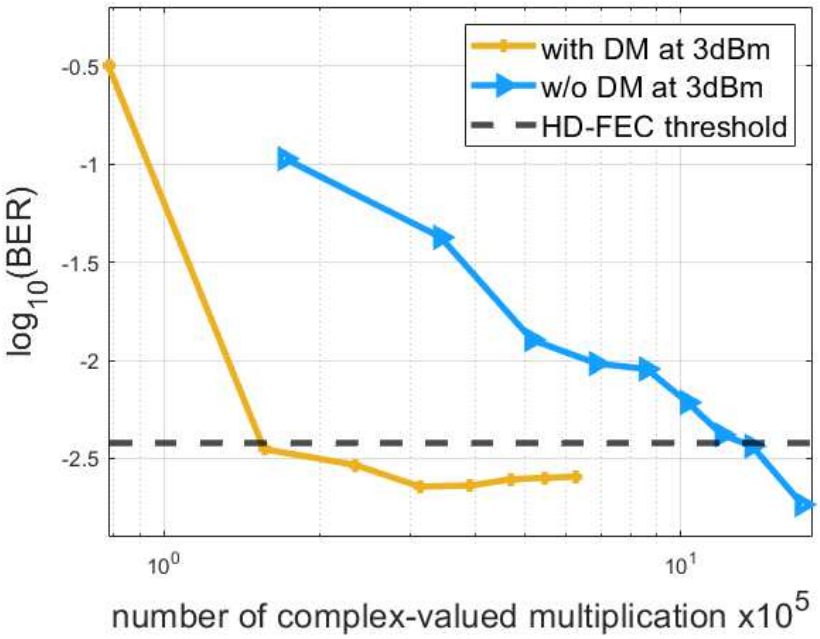

Figure 5.14: BER achieved for different CC values for two datasets where the dispersion-managed link is used and two where not (at 3 dBm)

The optimised NN models have once again undergone additional pruning to reduce complexity without significantly compromising performance. The accompanying graph 5.14 illustrates the relationship between the computational complexity of the pruned model and the system performance, expressed in BER.

From the figure, it can be seen the NN-based equalisers' performance for both links improves with increased computational power. Furthermore, it is possible to eliminate up to 80% of the weights at a launch power of 3 dBm for the dispersion-managed system and still remain below the HD-FEC cutoff. However, for the dispersion-unmanaged link, only 20% of the weights can be removed to keep the BER performance above the threshold. It can also be seen that the dispersion-managed link requires significantly less complex multiplications. This corresponds to the fact that the channel memory was reduced by the inclusion of DCF fibre. Finally, it can be seen that with the inclusion of dispersion management, the number of complex-valued multiplications per sample to keep the performance of the system below the threshold is reduced to almost 12% compared to uncompensated systems. This makes a clear case for the capability of the DM system to simplify the interaction of signal and noise during signal propagation, which is further mitigated through the use of NN equalisers.

5.3 Conclusion

In summary, the combination of optical solutions with NN-based equalisers has proven to effectively reduce the complexity of the equalisation process. Optical phase conjugation

offers the advantage of additional reducing fibre effects, while dispersion management techniques significantly decrease complexity while keeping the same performance. This is likely because dispersion management is highly efficient in eliminating dispersions throughout the link and reducing channel memory.

Although these techniques successfully reduce equaliser complexity, they require modifications to the link and the addition of extra components such as OPC or dispersion compensating fibres (DCFs) into the system. In the next section, some techniques will be explored for their potential to facilitate and enhance the learning process in Neural Networks.

Chapter 6

Techniques to improve the learning process in Neural Networks

Another way to reduce the computational load and required electricity in using ML-based equalisers is to make the learning stage more efficient. This can be achieved through several approaches. The effectiveness of learning can be increased by carefully preparing and enhancing the training data (data preprocessing). Techniques like data normalisation, data cleaning, data transforming, data augmentation, and data pruning enable training more effectively with fewer data points. Another way to speed up the learning process is by using pre-trained models or transfer learning techniques by starting with a model that has already learned relevant features from a similar dataset or task. By doing this, the equaliser may make use of the knowledge gained during earlier training phases and reduce the amount of time, resources, and data needed for computation and fine-tuning. By implementing these techniques, the learning stage of ML-based equalisers can become more efficient, either enabling faster training, training with fewer data points, or better utilisation of available training data. This, in turn, leads to reduced computational load and electricity requirements, making ML-based equalisers more practical and environmentally friendly in various applications.

This chapter investigates some of the techniques with the potential to improve learning. Each technique is introduced with a brief explanation, followed by a demonstration of its implementation and the resulting outcomes. The resulting performance of the equaliser and the impact on the learning curves are both considered in measuring the effectiveness of each technique.

Learning curves, as previously stated, are graphical representations of how a model's performance evolves over time, typically as the length of training (measured here as the number of epochs) increases. These curves are useful in determining the model's ability to learn from data, and they usually include two components: training loss and valida-

tion loss. The training loss evaluates the model's performance on the training dataset throughout the training process by measuring the difference between predicted and actual symbols (here, measured using MSE). Ideally, the training loss should decrease, indicating that the model is gradually improving its fit to the training data. Similarly, validation loss is a metric that evaluates the model's performance on a specific dataset known as the validation dataset. Unlike the training data, this set is not used during the training phase but is critical in determining how well the model generalises to new, previously unseen data. The validation loss is beneficial for detecting overfitting, which occurs when a model performs well on training data but fails when applied to new, unknown data [36]. In this chapter, to avoid potential overestimation of the suggested techniques' impact on learning, the focus shifts from training loss to validation loss as a more robust measure. This adjustment ensures a more accurate assessment of the techniques' effectiveness and their influence on the learning process.

6.1 Data pruning

Firstly, the decision was made to explore techniques that could facilitate faster training with fewer data points. One such technique is data pruning, which involves cleaning and optimising data sets by removing unnecessary, redundant or noisy training instances. The goal is to improve the quality and efficiency of data for modelling, analysis, and other uses. This should, in turn, increase the effectiveness of the learning process and decrease overfitting risk, which happens when the model learns to fit the training data too closely and fails to generalise to new, unseen data.

Data pruning might be particularly helpful when the learning process is computationally expensive or the training data is large. By reducing the size of the training dataset, data pruning can speed up and improve the learning process while still creating accurate models. However, pruning data should be done cautiously, as losing crucial training examples can negatively impact model performance. Many important factors need to be considered to reliably rectify input data for data pruning, including data quality evaluation, preprocessing strategies, and validation procedures. Assessing data quality involves finding and fixing errors, inconsistencies, and outliers to ensure accuracy. Furthermore, confirming the data is complete is essential, guaranteeing no missing values are in the dataset and choosing how to handle them, if any, including removal or imputation. Additionally, it is necessary to confirm that similar data is consistently represented throughout the dataset in order to maintain consistency. Every attribute and feature in the dataset is evaluated for relevance, and any redundant or unnecessary features that do not significantly advance the analysis or modelling process are eliminated.

There are various preprocessing strategies for implementing data pruning, depending on the specific data that needs to be removed (e.g. random sampling, feature selection, instance selection, etc.). Broadly speaking, these approaches can be categorised into three

types:

- Type 1- This type involves modifying the number of input samples by removing unnecessary points or outliers. The goal is to reduce the overall size of the dataset while preserving its essential characteristics.
- Type 2- In this type, the focus is on modifying the number of features in the dataset. Techniques such as feature selection are employed to identify and retain only the most relevant and informative features, discarding the rest. This helps reduce dimensionality and improve computational efficiency.
- Type 3- Type 3 pruning involves modifying the ranges of the input samples. This can be achieved by scaling or transforming the data in order to narrow down the range of values within each feature. Doing so can eliminate redundant values, leading to a more compact and focused dataset.

The method that is proposed in this section is a combination of type 2 and type 3 pruning techniques. This means that the suggested approach aims to simultaneously shrink the range of parameters within the dataset and discard any unused features. By narrowing the parameter ranges and effectively removing irrelevant features, the resulting dataset becomes more concise and manageable for further analysis or modelling.

A crucial step after applying specific preprocessing techniques is assessing how they affect the model's performance and making adjustments to the techniques as needed. As mentioned before, the impact of the data pruning technique on learning is measured using the validation loss (loss achieved using the validation set). Furthermore, the resulting BER serves as a measure of the model's overall effectiveness. As such, a thorough examination of both variables is necessary to determine the balance between facilitating the learning process and its impact on output.

QAM transmission constellations exhibit symmetry between about two axes and two diagonal lines, which maintains consistent energy levels for each signal point. This inherent symmetry provides an opportunity to leverage the separation of points within a single quadrant to predict the positions of points in the remaining quadrants. By examining a single quadrant of a QAM constellation, information on the separations of the clouds can be obtained. Then, the known relationships between the points within a single quadrant can be utilised to infer the positions of points in the other quadrants. This can simplify the data while preserving the necessary information for accurate equalisation.

To achieve this and expedite the NN equalisation process, a data-pruning technique called "folding" is suggested. This technique involves folding or mirroring the data along the axes. By leveraging the inherent symmetry in QAM transmission constellations and employing data pruning through folding, it is possible to streamline the training and testing procedures for NN equalisation. This approach results in faster processing times.

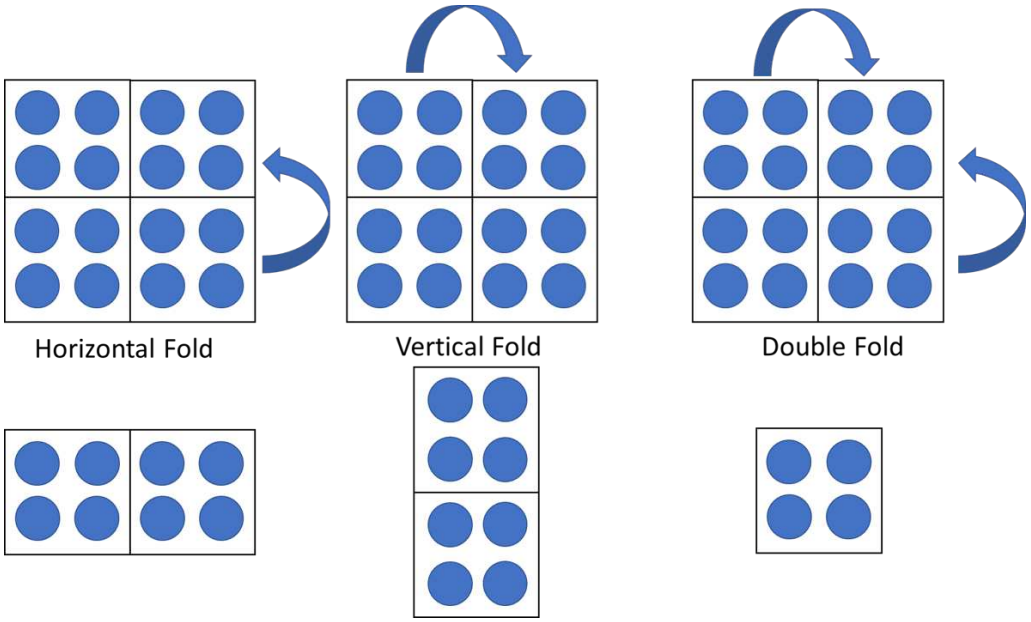

Figure 6.1: Modification of the dataset for data pruning using folding.

However, it is important to investigate the impact of the data simplification due to folding on the equaliser's performance, see Fig. 6.1.

6.1.1 Implementation of data pruning

The illustration of the folding procedure is demonstrated in Fig. 6.1. From there, it can be seen that data is "folded" across the x or y-axis depending on whether horizontal or vertical folding is applied. Additionally, double-folding is defined as an application of vertical folding to horizontally folded data.

During the folding process, a "flag" value is assigned to each data point, indicating whether it originates from the positive or negative side of the axis. This flag information is crucial for the unfolding procedure, where the equalised points are restored back to their original quadrants. The specific code modifications required to implement the fold and unfold operations are provided in the Appendix. The folding technique enables a compact representation of the QAM constellation by reducing redundant points and retaining the necessary information for accurate equalisation.

Figure 6.2 illustrates a schematic representation of the NN-based equalisation process with the folding. The procedure can be summarised as follows:

- Folding: The data undergoes folding, resulting in a compact representation of QAM constellations. This folding operation is applied to the symbols from both the X and Y polarisations.

- Data Preparation: The sequence of consecutive "folded" symbols from the X and Y polarisations are combined into a vector format. This vector size is $2(N_{taps}+1)$, and it serves as the input to the NNs, enabling the equalisation process.

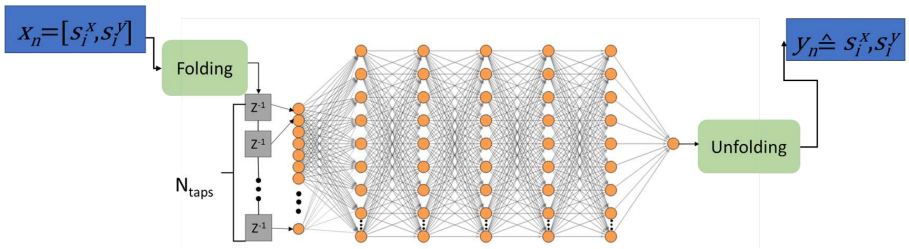

Figure 6.2: Modified structure of the equaliser for data pruning using folding.

- NN Processing: The folded symbol vector is fed into the NNs for equalisation. Here, a complex-valued NNs, defined in section 4.2.3, is implemented. This NN learns and adapts to the folded data representation, performing the necessary computations to enhance the accuracy of symbol recovery.

- Unfolding: Post-equalisation, the symbols are restored to their original form using the unfolding procedure. The stored "flag values" obtained during the folding process are utilised to reverse the folding and accurately reconstruct the symbols. After that, the unfolded symbols were used to assess the performance of the NN equaliser.

To investigate the effects of data folding on the NN-based equalisation, it was implemented to mitigate fibre nonlinearities in a 16-QAM 32-GBaud transmission link spanning a distance of 2000 km. For additional details about this communication system, refer to section 4.2.2.

As before, a dataset consisting of 2^{18} input samples was utilised for training the Neural Networks. The dataset was divided into 70% for training, 20% for validation, and 10% for testing purposes. Since the focus here is on investigating the impact of folding on the training or learning process, the equaliser with folding did not undergo separate optimisation. Instead, the previously obtained results from Bayesian optimisation for the NN structure, as described in section 4.2.4 and Table 4.1, were used.

To explore the influence of folding on the learning process, the results obtained from the data-pruned NN were compared with those from the normal (unpruned) NNs. This comparison provides insights into how data folding affects the performance and learning capabilities of the NN-based equaliser.

Figures 6.3 depict the impact of data pruning using the folding concept on the learning curves, as indicated by the validation loss and the resulting BER performance. The simplest dataset (double-fold) demonstrates the swiftest learning regarding validation loss. Overall, it is evident that data pruning leads to an improved and faster learning curve in terms of validation loss. However, fig.6.3 b) shows that this enhancement in the learning curve does not correspond to improved BER performance of the system. Notably, the BER performance of NNs implementing data pruning reaches saturation much earlier than standard NNs.

Furthermore, Figure 6.4 compares the resulting constellations between equalisation

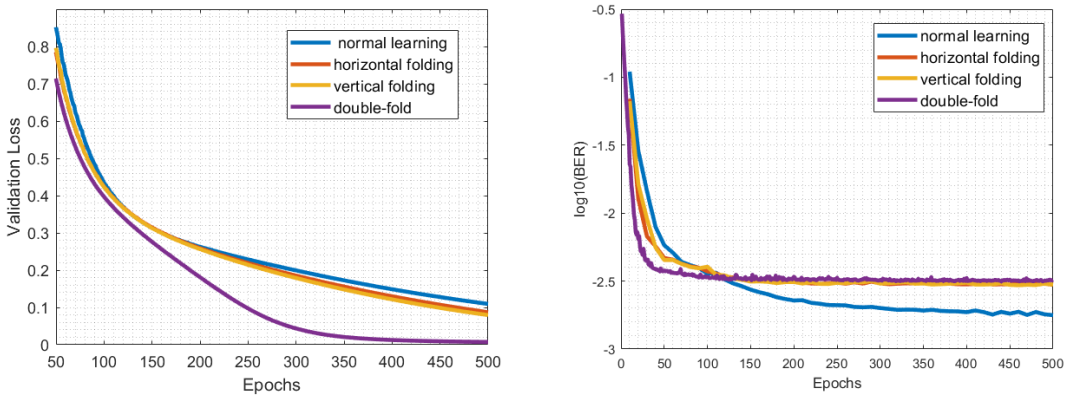

Figure 6.3: Comparison of the results for standard learning vs. data pruning using the folding concept for -1 dBm a) validation loss; b) resulting BER.

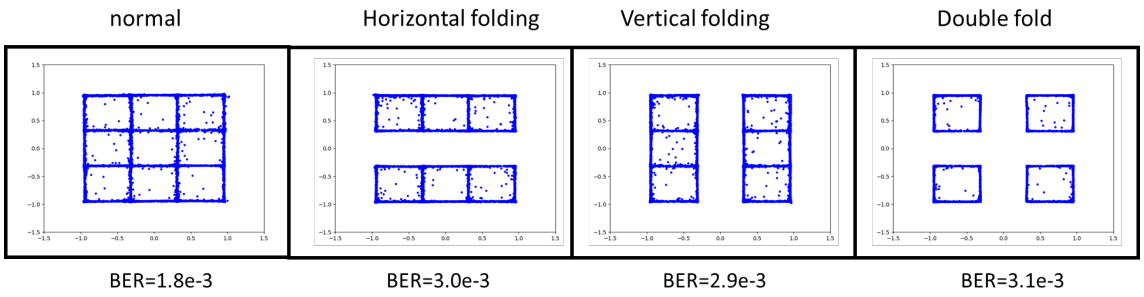

Figure 6.4: The resulting constellations from the application of data pruning using folding at optimum launch power.

based on NNs implementing data pruning and standard NNs. As expected, NN-based equalisation leads to "jail window" constellations, but depending on the folding method employed, the constellations may have missing borders where the folding was applied. Furthermore, the resulting BER for the data-pruned model is significantly worse than the unpruned one.

Based on the analysis, data pruning, through the implementation of data folding techniques, accelerates the learning process and simplifies it compared to the standard method. However, it is important to note that this approach significantly impacts the performance of the NN-based equaliser. One of the reasons for this impact is the loss of information associated with symbols that cross the boundaries created by the folding process. When folding the data, symbols that were originally positioned near the fold boundaries may now appear in different quadrants or regions of the folded constellation. This change in symbol positioning can introduce errors and distortions during the equalisation process. As a result, the accuracy of symbol recovery and the overall performance of the NN-based equaliser may be compromised when employing data folding techniques. The loss of information regarding symbols near fold boundaries impedes the equaliser's ability to accurately and effectively mitigate impairments, such as fibre nonlinearities, within the QAM transmission. Further research and optimisation are necessary to mitigate the limitations associated with information loss during the folding process, ensuring improved performance and enhanced robustness of the NN-based equaliser in QAM communication systems.

6.2 Curriculum learning

Another technique to accelerate learning is to use available data better. One such technique is curriculum learning, a machine-learning approach that involves presenting the training data to the learning algorithm in a specific order or curriculum intended to improve the learning process [99]. The goal of curriculum learning is to gradually increase the difficulty level of the training data, starting with easier data and progressively introducing more difficult data. By following a carefully designed curriculum, curriculum learning can improve performance without requiring additional computational resources. It achieves this by guiding the training of machine learning models in a meaningful and structured manner, as opposed to the conventional strategy of random data shuffling.

Curriculum learning techniques have been successfully applied in various machine learning tasks. However, there are certain considerations to bear in mind. One challenge lies in determining the optimal ordering of samples arranging them from simple to complex. Additionally, defining an appropriate pace function for presenting increasingly difficult information is crucial. These factors may limit the application of curriculum techniques, as careful consideration is required to establish an effective curriculum [100].

Given that the human brain serves as the model for neural network topologies, it makes

sense to assume that the learning process should similarly draw inspiration from human learning. Hence, the inspiration for curriculum learning originates from the observation that people typically learn systematically, with more straightforward concepts coming first. This method works well for learning various things, including language and motor skill training. Curriculum learning offers potential advantages that enhance the overall efficacy of the learning process. Firstly, it can increase the convergence speed during training, leading to quicker results. Secondly, it can yield higher accuracy levels, improving the quality of the outcomes. The issue of becoming stuck in local minima, where the algorithm learns to optimise a non-optimal solution because it has yet to be exposed to more complicated examples, can be avoided.

Curriculum learning can be put into practice in a number of different ways. Using a heuristic or pre-determined ordering of the training samples according to their difficulty is a popular strategy. Utilising an adaptable curriculum is a different strategy, whereby the examples are changed throughout the learning process based on the learner's performance.

The effective NN equaliser needs to learn the impact of the neighbouring symbols and the distortions caused by the noise. Curriculum learning can be employed to speed up the learning of the impact of consecutively transmitted symbols while also facilitating the understanding of noise effects. Gradually introducing increasingly complex datasets during training can achieve a faster learning curve than the standard learning approach for neural networks.

By employing curriculum learning, the NNs are trained in a more controlled and structured manner. The initial exposure to simpler datasets helps establish a solid foundation and quickly grasp the fundamental relationships between transmitted symbols and their impact on the received signals. As the NNs demonstrate proficiency in handling simpler datasets, more challenging datasets can be gradually introduced. These datasets incorporate an increasing impact of noise. As the complexity of the datasets increases, the NNs can leverage their prior knowledge better to understand the impact of noise, resulting in faster adaptation and improved equalisation performance.

In the next part, the suggested implementation of the curriculum learning is shown to equalise the signal received for the 16 QAM 32 GBaud optical communication transmission over 2000 km. This dataset is described in more detail in section 4.2.2. The implementation of curriculum learning utilised the NNs equaliser, which is explained in section 4.2.3. Since the objective was to examine how curriculum affects learning, it was not independently optimised. Instead, the previously achieved results from optimising the NN structure in Table 4.1 were employed.

6.2.1 Implementation of curriculum Learning

The simulation dataset utilised to demonstrate the functioning of curriculum learning was sourced from section 4.2.2. For the practical application of curriculum learning, it was

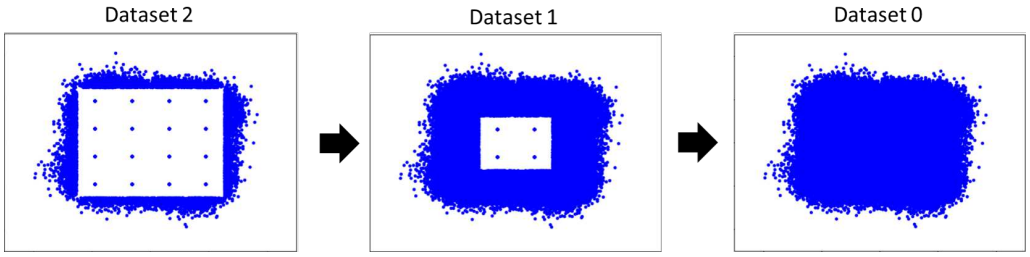

Figure 6.5: Modification of the dataset for the curriculum learning from "easy to difficult".

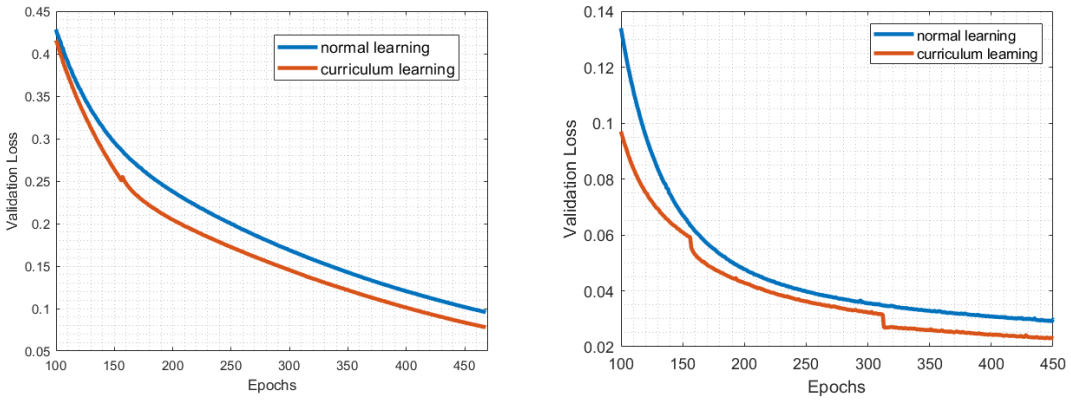

Figure 6.6: Comparison of the results for standard learning vs curriculum learning in terms of validation loss for two launch power: a)2 dBm and b)4 dBm

essential to prepare a dataset that transitioned from 'easy' to 'difficult' stages. The initial step involved condensing all the received constellations to the nearest points, signifying the QAM symbols. Subsequently, a similar dataset was created, where all constellations, except for the outer border ones, were condensed. This process was iteratively performed until the final dataset was left unaltered, representing the highest complexity. To use such an approach, the Gaussian (circular) representation of the noise is assumed. The set of datasets employed for the 16 QAM system is visually depicted in fig. 6.5. After that, for each dataset, the vector of N_{taps} consecutive symbols from both X and Y polarisations were combined to be used as input to the NNs.

The curriculum learning process is implemented as follows: Initially, the model is trained using the "simplest" dataset (Dataset 2 from Fig. 6.5) to establish a strong foundation. As the training progresses and the NNs begin to saturate in performance on the current dataset, the curriculum learning switches to a more complicated dataset. This iterative process continues until the model is trained with the original "complex" dataset. Finally, the performance of the trained equaliser is evaluated solely on the standard dataset.

Figures 6.6 provide an analysis of the learning processes, examining validation loss between standard and curriculum learning approaches at two different launch power values. In Figures 6.6, it is evident that there is a slight jump in validation loss around 150 epochs, indicating a point where the dataset transitioned from "easy" to "more complex.". However, while the first dataset change at 150 epochs resulted in fast decreases in the loss,

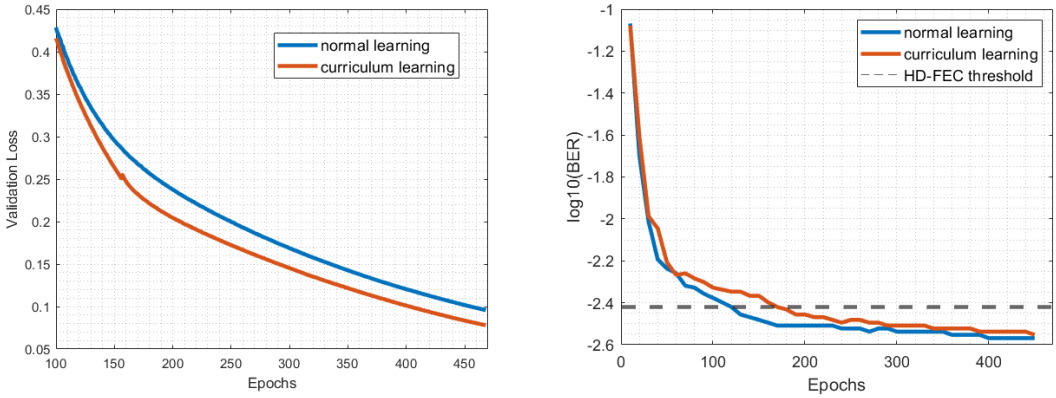

Figure 6.7: Comparison of the results for standard learning vs curriculum learning for 2 dBm in terms of a) validation loss; b) BER

the second dataset change predominantly influenced the loss function with a less visible improvement in BER. Comparing the results between normal and curriculum learning confirms that curriculum learning accelerates the learning process and yields better validation loss metrics.

Figures 6.7 demonstrate analysis of the learning process and performance between standard and curriculum learning approaches at 2 dBm. However, while curriculum learning resulted in reduced validation loss, it did not necessarily translate into an overall improvement in the BER system's performance (see 6.7b). Contrary to expectations, curriculum learning led to a slight decrease in the BER.

In summary, the impact of curriculum learning on the learning process was investigated by comparing the validation loss and BER performance between standard and curriculum learning approaches. The results demonstrate that curriculum learning accelerates the learning process and improves validation loss metrics. However, this came at the expense of slightly reduced BER performance.

6.3 Multi-output Neural Networks

Another approach to enhancing the learning process further involves leveraging multi-output neural networks to extract more information from the available data.

A multi-output neural network is a type of artificial neural network capable of generating multiple output values for a given input. Unlike conventional neural networks that produce a single output, multi-output networks are designed to handle situations where a given input may require more than one output. These networks find applications in various domains, such as image segmentation, object detection, and natural language processing. For instance, in image segmentation, the network predicts object boundaries and object classes as separate outputs, enabling more detailed analysis and understanding.

To create a multi-output neural network, the final layer of the network is expanded with additional output nodes. Each output node corresponds to a separate output value

that the neural network aims to predict. During training, the network is tuned to minimise the discrepancy between the expected and actual output values for each output node.

In the context of an optical communication system, it is well-known that neighbouring symbols can influence each other. Therefore, incorporating multi-symbol outputs can provide significant advantages for the learning process. By considering multiple symbols in the outputs, the cost function used for learning can be adjusted to accommodate the equalisation of multiple symbols. In this particular study, the decision was made to initially test the system's response to equalising three symbol outputs.

By adopting a multi-output neural network approach and considering multiple symbol outputs, the learning process can potentially benefit from capturing more intricate dependencies and patterns within the data. Furthermore, as this NN result in multiple-symbol equalisation, it introduces a possibility of more effective decision-making in the following stages of detection.

Consequently, the NN become more adept at extracting nuanced relationships between input features and multiple desired outputs, resulting in enhanced predictive capabilities and a greater understanding of the underlying data.

6.3.1 Implementation of multi-output Neural Networks

To demonstrate the application of multi-output symbol equalisation, a dataset from an optical fibre link of the length of 2000 km, with the 16 QAM signal generated at 32-GBaud, was used (more in section 4.2.2). For training the NN, 2^{18} input signals were used at each launch power, divided as 70% training, 20% validation and 10% testing. The input for the neural networks was constructed by combining a sequence of N_{taps} of consecutive symbols from both the X and Y polarisations. Except for the final layer, cartesian relu was used as an activation function. The values for the epochs and the batch size are the same as before, 500 and 2000, respectively, Here, three-symbol output equalisation has been considered. The code was, hence, modified so that the Neural Network would be able to predict three symbols at the time. However, out of those symbols, only the middle one is of interest in our experiment. Bayesian optimisation was implemented to determine the optimal values for each value of the launch power. The results of the equalisation in terms of BER are presented in Fig. 6.9 and compared to the normal NNs (single output).

In addition to improving the learning process, it was crucial to investigate the potential enhancement in BER by modifying the cost function. To accomplish this, optimised structures were defined for each launch power value.

Figure 6.8 demonstrates the comparison of the validation loss function obtained between normal learning and three-symbol output. From there, it can be seen that the inclusion of the information of the output symbols provided a clear improvement in terms of the learning process.

Figure 6.9 provides insights into the BER performance of single-output Neural Net-

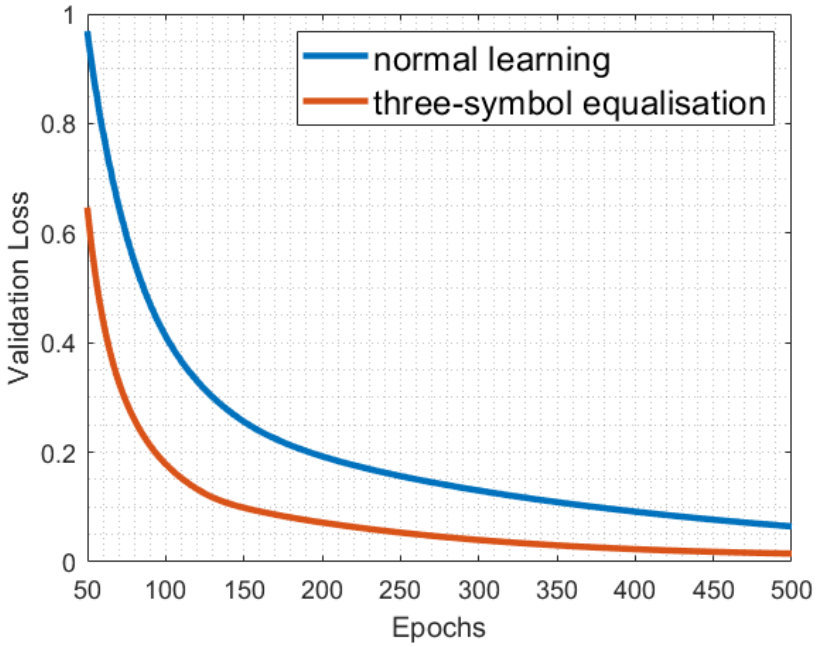

Figure 6.8: Comparison of the validation loss for standard learning vs. multi-output neural network at 3 dBm.

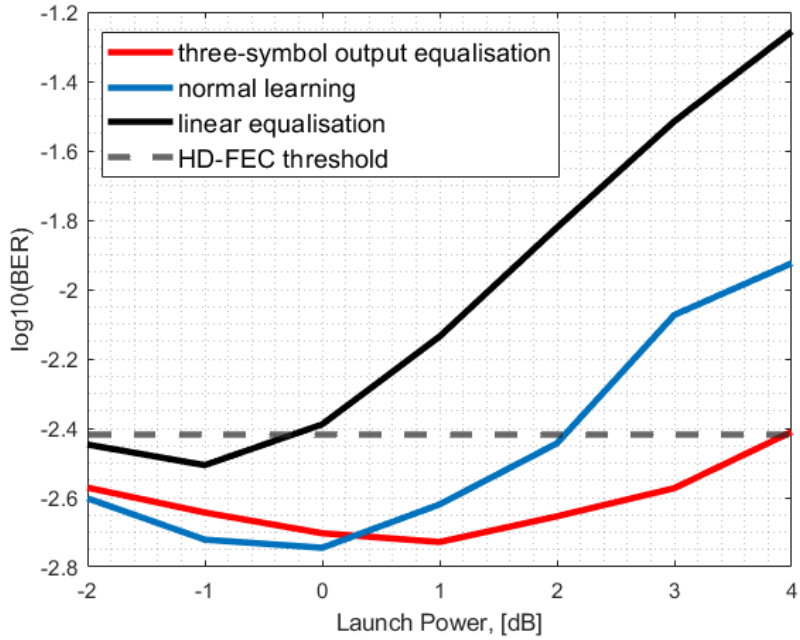

Figure 6.9: BER performance as a function of Launch Power for standard learning vs. multi-output neural network.

works (NNs) compared to three-symbol output equalisation. In the "linear regime," the results indicate a slight advantage for single-output NNs. However, as the launch power levels increase, the utilisation of three-symbol output equalisation through NNs exhibits superior performance. Notably, this approach significantly enhances system performance at power levels 3 and 4 dBm below the HD-FEC threshold. It is worth noting that the impact of three-symbol training becomes more pronounced at higher power levels. This observation suggests that the multi-output NN approach excels in capturing and addressing nonlinearities within the data. This may be due to the fact that at higher powers, the noise is deterministic, and the effect of the neighbouring symbols on the equalisation is higher. This also corresponds to the fact that at higher powers, more number of $N_{taps}$is required, further showcasing the increased influence of the neighbouring symbols at the nonlinear regime.

The utilisation of three symbols as output has been shown to enhance the learning and the BER performance of equalisation. A further test would be to investigate the potential to utilise multi-symbol equalisation with the soft-decision encoder in the simplest way by means of averaging between multiple predictions corresponding to the same symbol.

6.4 Conclusion

The chapter explores techniques to improve the efficiency of the learning stage in ML-based equalisers. One technique explored was data pruning, which was implemented by the use of the folding technique. The results of the implementation suggest that while data pruning enables faster and simplified learning compared to the standard method, it significantly affects the performance of the NN-based equaliser. This impact may be attributed to the loss of information regarding symbols that overlap the fold boundaries when employing data folding techniques.

Another technique, curriculum learning, presents training data to the learning algorithm in a specific order, gradually increasing the difficulty level of examples. This approach guides the training process in a structured manner and can accelerate learning without requiring additional computational resources. The results of the implementation of curriculum learning have shown its potential to speed up learning. This comes at the cost of a slight decrease in BER. However, the effectiveness of curriculum learning might be limited to low noise power regimes.

The use of multi-output neural networks, capable of generating multiple output values from a single input, was also investigated. By considering multiple symbols in the outputs, the learning process can capture more intricate dependencies and patterns within the data. The findings suggest that multi-output neural networks can enhance performance, especially in high launch power levels, by modifying the cost function to accommodate multi-output scenarios.

Chapter 7

Crowd Equalisation

Another way to address the computational complexity challenge of machine learning-based equalisers is an analogue implementation of them in the electrical or optical domain [91; 101; 102]. Optical platforms may be significant in the future of computing because of their inherent advantages of high bandwidth and low power usage. A nonlinear activation node is placed after layers of matrix multiplication in the optical application of a structured NN. This sort of processor can perform complex high-speed ML computations with potentially low power and noise floor, according to [91]. This hypothetical low-noise limit has not yet been put into practice, though. Nonlinear activation nodes are typically implemented using active devices, which introduce noise into the computation chain and result in a cascade effect that is currently the bottleneck of the optical application of NNs [102; 103]. In deeper neural networks (DNNs), which have more layers and neurons, this impact is more pronounced. This, in fact, suggests designing smaller neural networks. Designing smaller NNs or breaking down large ones provides the natural benefit of reducing the complexity of the hardware. Numerous optical solutions involve performing the nonlinear activation function and implementing the multiplication step of the NN using photonic components [104]. The literature is filled with suggestions for the optical implementation of NNs, most of which are for a few layers. For some impressive examples, refer to [105; 106].

This chapter offers a straightforward answer to this problem by building a bunch of low-complexity NNs and using them to equalise the signal. The goal is to demonstrate how using independently trained NNs can enhance decision-making. For that, the committee learning approach has been investigated.

7.1 Principle of Committee Learning

Committee learning is a technique in machine learning that involves training multiple models on the same dataset and then combining their predictions to make a final prediction. This technique is often used to improve the accuracy and reliability of machine learning models. The basic idea behind committee learning is that different models may have different strengths and weaknesses, and by combining their predictions, a more ac-

curate and robust model can be created. This is similar to the idea of "wisdom of the crowd," according to which the collective judgement of a diverse independent group of people performs better than that of an expert.

Using a committee machine offers several advantages in the context of machine learning. Firstly, it assists in preventing overfitting, a problem that occurs when a model is too closely fitted to the training data, leading to inadequate performance on new data. A committee machine efficiently reduces overfitting by combining predictions from various models. Secondly, it improves prediction accuracy by using the various advantages that each individual model possesses. The incorporation of multiple perspectives and approaches can contribute to more robust and reliable predictions. Finally, a committee machine may speed up the machine learning model's training process. This acceleration can be made possible by cutting down on the total amount of time needed for model training by parallelising the training phase for each individual model.

There are several ways to combine the predictions of multiple models in committee learning. One common approach is to use a simple voting scheme, where each model makes a prediction, and the final prediction is based on the majority vote. Another method is to use weighted voting, where each model's prediction is given a weight based on its performance on a validation set. Committee learning can be used with various machine learning models, including decision trees, neural networks, and support vector machines. It is beneficial when the dataset is noisy or highly uncertain, as combining multiple models can help reduce these issues. This chapter shows that various configurations can be used to take the opinion of the crowd of NNs into account and argue that this impact can be explained by the eminent presence of a random element and also the impact of additive noise at each nonlinear node [107].

7.2 Implementation of crowd equalisation

Using the principle of committee learning, a crowd of individual equalisers is formed as small NNs, each capable of marginally improving the performance of the optical communication system by equalising the received signal. This "crowd of equalisers" is supposed to tackle the "randomness" factor of the output of each of these individual equalisers. The weighted average of the outputs of the crowd members will form the final equalised symbol. Therefore, individuals are trained to equalise the input symbols with no knowledge about further averaging their output. There are two questions to answer: i) how to make individual equalisers linearly independent, and ii) how to select the crowd members.

To answer i), it is important to consider that the output of a multilayer perceptron, as our individual NN of choice, is the result of a series of nonlinear transformations. This means that before the final linear layer, two equalisers must differ in their weights, design, or activation functions in order for them to be linearly independent. Due to the regression nature of the problem, the final activation function of all these NNs is a linear function.

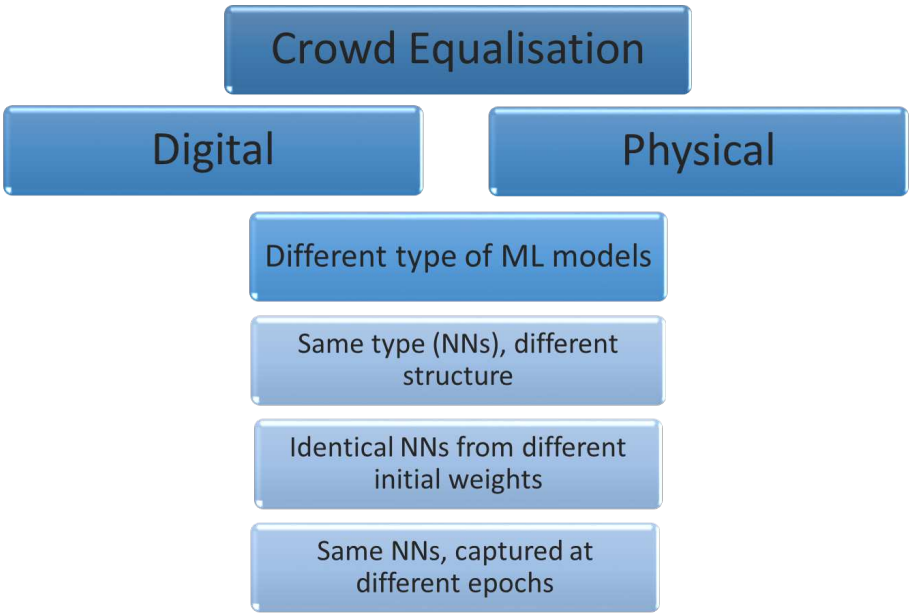

Figure 7.1: Different ways crowd equalisation can be combined

As a result, two NNs with various numbers of layers or neurons are likely independent of one another. Additionally, it can be hypothesised that even a slight (but significant) random variation in the link weight values, particularly in the first layers (closer to the input layer), is sufficient to transform two identical NNs into linear independent ones. There are several ways to ensure that the crowd members are linearly independent, such as having different weights, architecture or even activation functions.

Different ways have been considered for incorporating the crowd's collective wisdom, as can be seen in Fig. 7.1. Here, crowd equalisation in both the digital and physical domains is considered. The main difference is that the implementation of NNs in the physical domain would result in the presence of noise within the system—more on this in the following sections. The selection of the crowd's members depends on how the crowd was implemented (Fig. 7.1). Overall, a selected number of equalisers are individually trained using training/validation sets. After that, the trained models are saved and applied to the testing set. As was mentioned, it is important that the models used should be linearly uncorrelated. Based on the model's performance in the testing stage, the top candidates are then selected to form a crowd of various sizes. The performance of the crowd is then estimated using the "crowd-testing" dataset, which is not included in the training or testing stages of the individual NNs. The combined predicted output is a weighted sum of the outputs, giving more weight to more accurate individuals in the testing set. Therefore, the output of the equaliser is:

$$y = \frac{1}{\sum_{i \in \mathcal{C}} BER_i\left(\mathbf{x}^{\text{test}}\right)} \sum_{i \in \mathcal{C}} \frac{f_i\left(\mathbf{x}^{\text{crowd test}}\right)}{BER_i\left(\mathbf{x}^{\text{test}}\right)} \tag{7.1}$$

Where i is an individual in the crowd $\mathcal{C}$, $\mathbf{x}^{\text{test}}$ and $\mathbf{x}^{\text{crowd test}}$ are the input feature vector

drawn from the test and crowd test set, respectively, $f_i(\mathbf{x})$ is the output of the ith individual, and $BER_i(\mathbf{x})$ is the BER of the ith individual. In Eq. (7.1), y is the QAM symbol output of the equaliser attributed to the middle symbol in $\mathbf{x}^{\text{crowd test}}$. This y is then used in a minimum distance detector to detect bits.

7.3 Physical implementation of Crowd Equalisation

As mentioned, the notable characteristic of physical implementations of neural networks is the presence of noise within the system. This noise can arise from various sources, such as electrical fluctuations, thermal effects, or imperfections in the hardware components. The introduction of noise in the physical implementation of neural networks can significantly impact their performance and behaviour. It can cause inaccuracies in the computation, distort the signals transmitted between neurons, and potentially affect the overall reliability and robustness of the network. Consequently, dealing with and mitigating the effects of noise becomes an important consideration in designing and optimising physical neural network implementations.

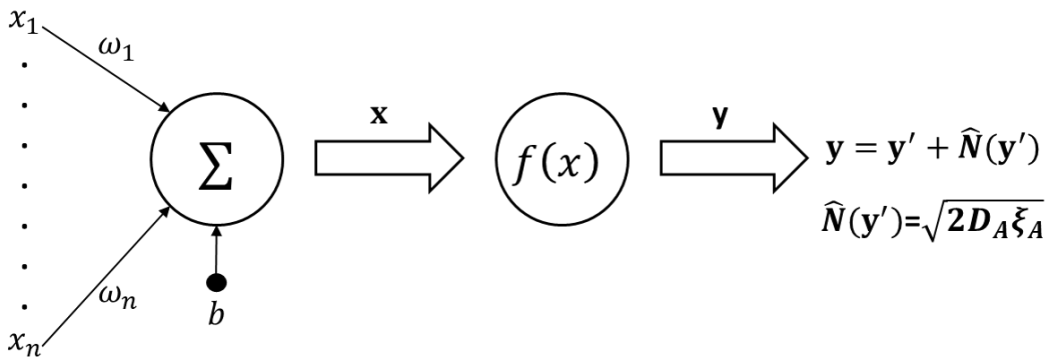

Figure 7.2: A structure of simple neuron with added noise

This section explores whether employing crowd equalisation in the physical implementation of neural networks enhances their resilience to noise. For that, NNs are considered with a noisy implementation, i.e. when matrix multiplication or activation function operations introduce noise and consequently reduce the Signal-to-Noise Ratio (SNR). Here, only noisy activation functions are considered, assuming the matrix multiplication is ideal. The SNR is defined as the ratio of the average of the output of the activation function to its variance [107].

Considering noise in the training process has been shown to improve resilience towards the ubiquitous noise and performance [108; 109; 110]. Furthermore, noise is present in any implementation of the NN; therefore, a more realistic assumption is to include it in the training stage.

The following subsections demonstrate the results of some of the implementations mentioned above from 7.1of crowd equalisation.

7.3.1 Crowd with different configurations of Neural Networks

In this section, the members of a crowd equalisation consist of Neural Networks (specifically, complex-valued MLPs) with different architectures. NNs with different structures can capture different features of the input data. This is handy when dealing with complicated interactions between temporal samples of the input, such as an optical communication signal. Specifically, for this case, six different NNs with different sizes and forms are trained (see Fig. 7.3).

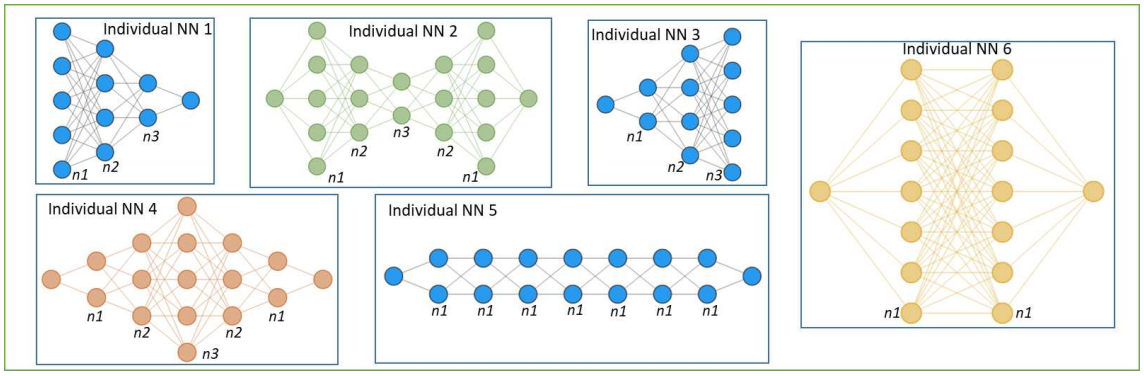

Figure 7.3: Configurations of individual NNs considered in the training of the crowd with different structures. The number of neurons in each layer of individual NNs changes accordingly so that the CC equals that of the single NN divided by the crowd size.

In this section, a large NN is divided into smaller ones to be used as crowd members. For that, in all simulations, the size of each member of the crowd is limited to keep the total computational complexity (in terms of the number of real-valued multiplications) of the equaliser fixed (see Table 7.1). The complex-valued MLPs have been implemented, minimising the complex-valued mean-squared error loss employing the Adam optimiser. The configuration of a single large NN and the structure of the input feature vector to each of the NNs is the same as in section 4.2.3. The dataset consists of experimental data containing transmitted and received 64 QAM symbols in a 28 GBaud dual-polarisation optical communication system of length 400 km; for more details about the experimental setup, see 5.1.1. The dataset is divided into training (70%), validation (10%), testing (10%) and crowd-testing (10%). The last set is only used to test the performance of the crowd and is not included in the training or testing stages of the individual NNs. After training six individuals, the best two, three, and four are selected to form crowds of sizes two, three, and four, respectively, having the full models saved. Then, these models are used on the crowd test set to calculate the combined predicted output using Eq. 7.1. The noise added to the output of each neuron is a complex-valued Gaussian random variable with zero mean and different variance levels. Figure 7.4 shows the achieved BER (calculated as the number of mismatches of the transmitted and received bits) versus the SNR for a single equaliser, crowds of two, three, and four. The single equaliser performance significantly deteriorates as the noise power increases. As is shown, the sensitivity of these equalisers'

Table 7.1: Complexity comparison between crowds of different size, when each of NNs has different configuration

Equaliser	Configurations of individual NN equalisers	total CC
Single NN	$[600, 600, 600, 600]$	4.3 mil
individuals in crowd of 2	$[900, 476, 223], [732, 732]$	4.3 mil
individuals in crowd of 3	$[746, 386, 174], [597, 597], [175, 389, 750]$	4.3 mil
individuals in crowd of 4	$[678, 335, 114], [517, 517], [119, 336, 684], [128, 236, 445, 236, 128]$	4.3 mil

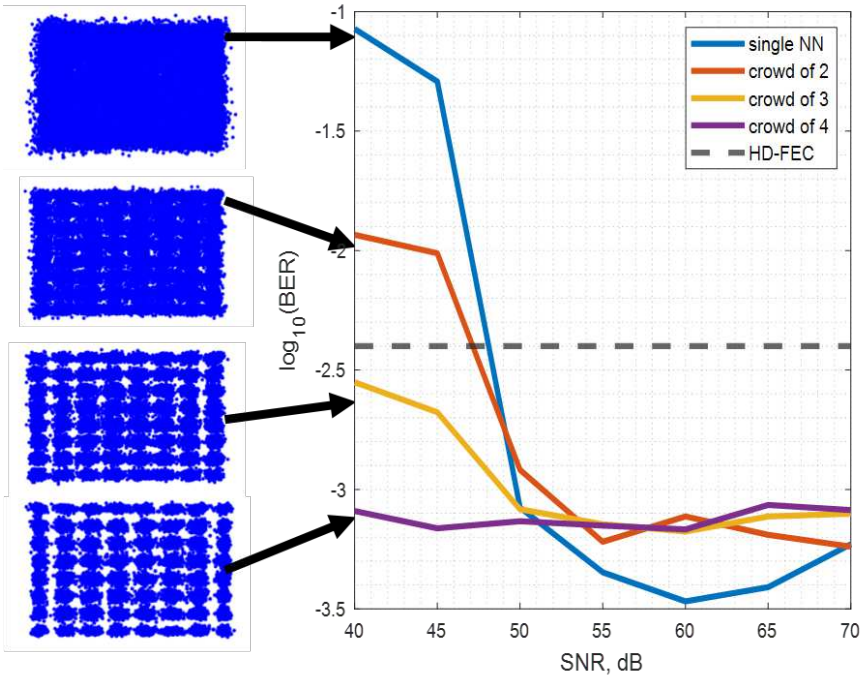

Figure 7.4: Resulted BER for the cases of the single NN vs combined (crowd) NNs, when SNR is between 40 and 70 dB, when crowd consists of NNs with different structures.

performance decreases as the crowd's size increases.

The constellations produced due to equalisation are shown on the left side of Fig. 7.4, clearly demonstrating an increase in noise reduction from the received symbols. As can also be seen in this figure, at high noise powers, a larger crowd results in a lower BER, particularly at SNR ≤ 50 dB. The improvement in cloud separation in the received constellation due to adding more people to the crowd, for the instance of the crowd of four, is shown in Fig. 7.10. This again indicates that the nature of the thing removed by averaging is noise.

7.3.2 Crowd with the same configuration, different initialisation

In this section, the crowd comprises NNs with the same configuration but with a different initialisation of the weights. In this case, there are two ways of deciding on crowd members. The first one is that each member will be an exact copy of the original NN found through Bayesian optimisation. In this case, the combined computational complexity will increase

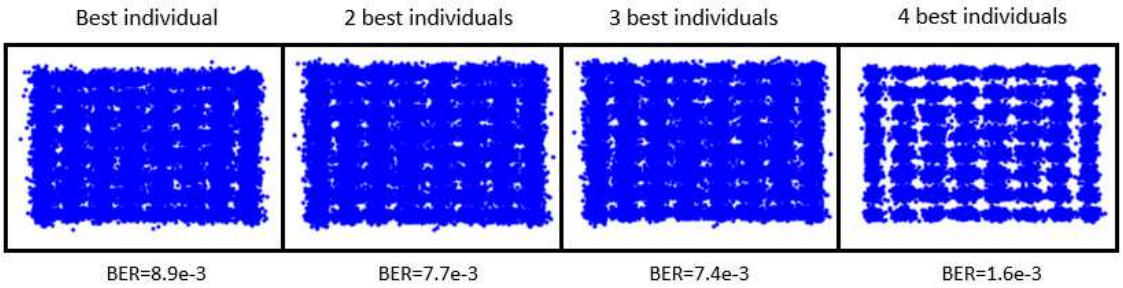

Figure 7.5: Improvement in BER and constellations for a crowd of 4 NNs when averaging takes place over 1, 2, 3, 4 best individual NNs when each of those NNs has different configuration

with the inclusion of each crowd member. The second way is to limit the size of each crowd member to keep the total CC (in terms of the number of complex-valued multiplications) of the equaliser fixed; see Table 7.1. For this, the shape of each crowd member was chosen to be a simple multilayer perceptron with two hidden layers with the same number of neurons per layer. In both cases, each crowd member was trained accordingly with different weight initialisation, and the best performers were chosen to form a crowd equaliser.

Similarly, the Adam optimiser was chosen for the backpropagation with a learning rate of 10^{-5} to minimise the complex-valued mean-squared error loss. The configuration of a single large NN and the structure of the input feature vector to each of the NNs is the same as before, where several QAM symbols before and after the symbol of interest are considered in each run. The data set consists of simulated data containing transmitted and received QAM symbols in an optical communication system of length 2000 km (see Fig. 4.5). The setup consists of a polarisation division multiplexing 32-Gbaud 16QAM transmitter, 20 spans of SSMF ($\alpha = 0.2$ dB/km, $D = 17$ ps/nm/km, $\gamma = -1.3$/W/km) and a coherent receiver with no nonlinearity compensation DSP, see 4.2.2 for details.

The whole data set was divided as before 7.3.1 into four parts: training (70%), validation (10%), testing (10%) and crowd testing (10%). The output of the crowd equalisation is also calculated with Eq. 7.1.

7.3.2.1 When combined computational complexity is the same

Similar to the technique from section 7.3.1, the large Neural Network is divided into smaller ones while maintaining the total computational complexity. However, implementation is simpler since it does not require as much "guessing" about how to choose each crowd member. Specifically, in this experiment, the number of layers of each member is kept at 2, which would aid in the optical realisation of neural networks by reducing noise. The table 7.2 showcases the comparison of the complexities between crowds of different sizes.

The results of the implementation of the crowd equalisation, where each crowd member is the same NNs, but with different initialisation, on the dataset, from 4.2.2 is provided

Table 7.2: Complexity comparison between crowds of different sizes when crowd members are the same and total CC stays the same

Equaliser	Configurations of individual NN equalisers	total CC
Single NN	$[800, 800, 50, 50, 800, 800]$	1.38 mil
individuals in crowd of 2	$[825, 825], [825, 825]$	1.38 mil
individuals in crowd of 3	$[673, 673], [673, 673], [673, 673]$	1.38 mil
individuals in crowd of 4	$[583, 583], [583, 583], [583, 583], [583, 583]$	1.38 mil
individuals in crowd of 5	$[520, 520], [520, 520], [520, 520], [520, 520], [520, 520]$	1.38 mil

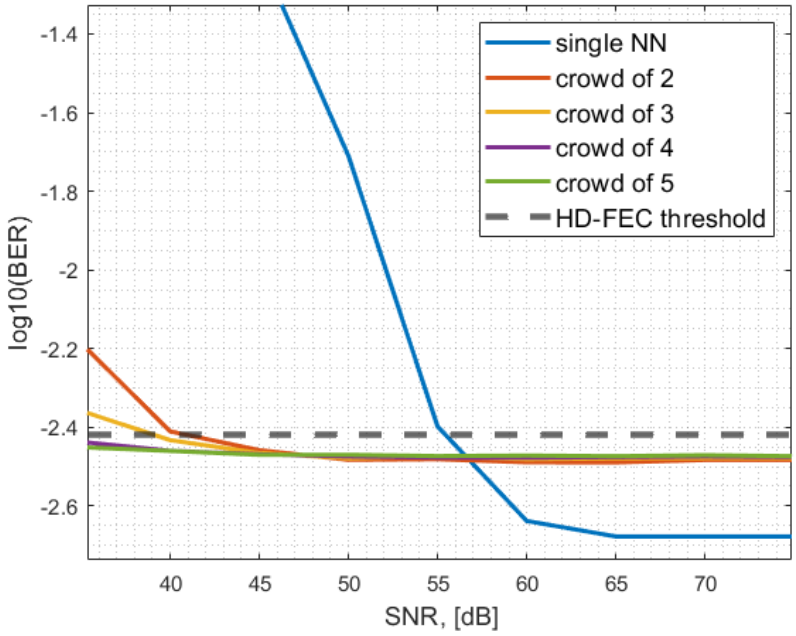

Figure 7.6: Resulted BER when SNR is between 35 and 100 dB at the optimum launch power for the cases of the single NN vs combined (crowd) NNs, consisting of the same NNs when total CC stays constant

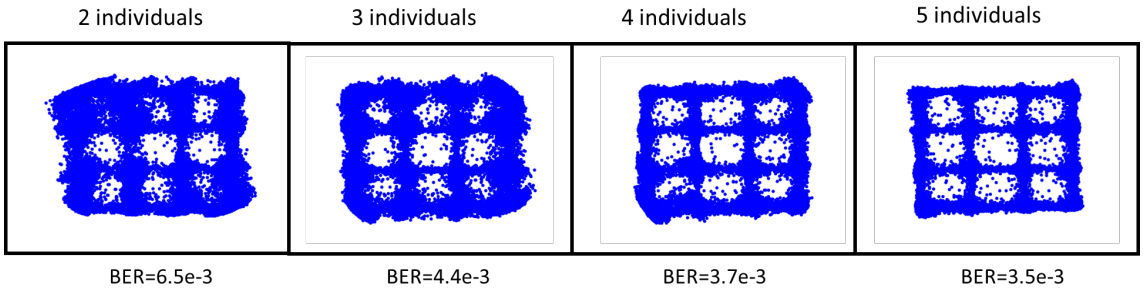

Figure 7.7: Improvement in BER for a crowd of 5 NNs at Launch Power of 0 dBm when averaging takes place over 1, 2, 3, 4, 5 best individual NNs, which have the same configuration but different initialisation(overall CC is kept the same).

in Fig. 7.6. From fig. 7.6, it can be seen that there is not much variation in the BER performance with the inclusion of more crowd members, and weighted averaging does not improve the performance of the system by much margin. Nonetheless, it can be seen that for the low values of SNR (lower than 55 dB), the inclusion of crowd equalisation improves the sensitivity of the NNs to the noise. Furthermore, it can be seen that using crowd equalisation (crowd of 5), the system's performance stays below the HD-FEC threshold throughout the range of SNR values of $35 - 100$dB. Figure 7.7 demonstrates the improvement in the constellations, as more members of the crowd are included for the equalisation.

7.3.2.2 When combined computational complexity is different.

In this configuration, crowd equalisation is employed, where each member has the same structure as the one obtained by Bayesian optimisation from section 4.2.4 and Table 4.1. As a result, the computational complexity of such a crowd equaliser is not identical to that of an individual equaliser but equal to $CC_{crowd} = N * CC_{indiv}$. This experiment aims to investigate whether the performance of a crowd equaliser can surpass that of a single equaliser that has been optimised solely through Bayesian optimisation. By leveraging the collective intelligence of the crowd members, the aim is to determine if further enhancements can be achieved beyond what has been achieved through Bayesian optimisation alone.

The configuration is tested on the same dataset from the 16 QAM 32 GBaud transmission over 2000 km fibre link from section 4.2.2, with individual neural network equalisers that are optimised in section 4.2.3, and 4.2.4. The crowd output is also computed using the same formula as Eq. 7.1.

Figure 7.8 depicts the achieved BER as a function of SNR for a single equaliser, for various configurations: a single equaliser, crowds consisting of two, three, four, and five equalisers. Both plots demonstrate a significant performance degradation for the single equaliser as the noise power increases. Also, the sensitivity to noise of these equalisers' performance decreases as the crowd's size increases. The performance of the crowd of five at 100 dB SNR surpasses that of the single equaliser in the nonlinear regime (Fig.7.8b at 4 dBm). However, in the "noisy" regime, the performance of a single equaliser is almost the same as that of a crowd at high SNR values (see Fig. 7.8a). This finding indicates that crowd equalisation, besides improving sensitivity to noise, can enhance equaliser performance beyond Bayesian optimisation.

Figure 7.9 illustrates the BER plotted against the launch power at the mid-SNR level of noise. The results demonstrate that employing a larger crowd size improves BER performance. It also shows that at mid-SNR values, the crowd equalisation approach delivers enhancements across all specified ranges of launch power. This observation further highlights the potential of crowd equalisation in achieving better performance.

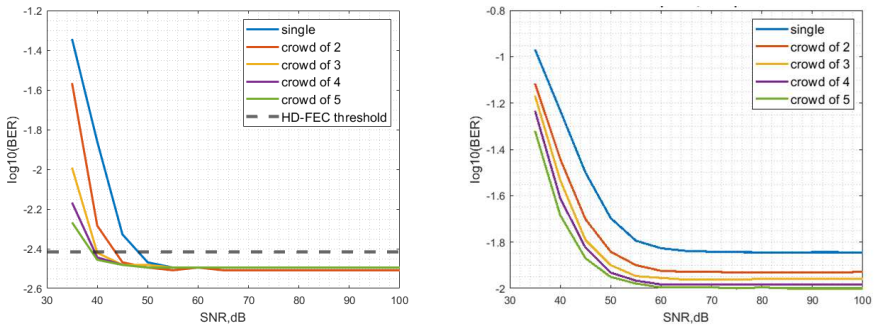

Figure 7.8: Resulted BER when SNR is between 35 and 100 dB at the optimum launch power for the cases of the single NN vs combined (crowd) NNs, consisting of the same NNs when total CC increases with each member of the crowd. Two launch powers are investigated: a)-2 dBm; and b) 4 dBm

Figure 7.10 illustrates how adding additional members to the crowd, for the case of the crowd of five, leads to an improvement in the separation of clouds in the received constellation.

Comparing the results of two approaches for crowd equalisation, using neural networks (NNs) with the same configuration but different initialisation, reveals interesting observations. It is evident that when the overall CC remains constant, the performance of the crowd equaliser excels in noisy regions (SNR lower than 55 dB) compared to a single equaliser. This improvement can be attributed to selecting NNs with fewer layers for the crowd, thereby minimising the impact of noise. However, as the SNR increases, the performance of a single equaliser surpasses that of the crowd equalisation with the same total CC as the larger NN is more capable of capturing the evolution of signal in the fibre.

Notable differences arise when employing NNs identical to a single optimised equaliser for crowd equalisation. At low SNR levels, the crowd implementation provides less BER improvement than the crowd equaliser with the same total CC. However, as the launch powers increase into the nonlinear regime, this crowd implementation has delivered improvements across all SNR values. Since it provides an improvement at 100-SNR, it means the performance of the crowd equaliser managed to achieve an improvement beyond the Bayesian optimisation. This can be potentially used to improve the performance of the equalisers but would require careful consideration of the required computational resources.

7.3.3 Crowds with the same configuration at different epochs

In this section, the "crowd" refers to members derived from the same model configuration, acquired via Bayesian optimisation, captured at different epochs instances. This approach necessitates the least amount of training relative to others; a single training session is required, and the crowd members are stored at specific, evenly spaced epochs at the end of an epoch cycle. These snapshots of the NNs are expected to exhibit linear independence. This is due to the fact that the weights of the NNs are likely to differ between epochs,

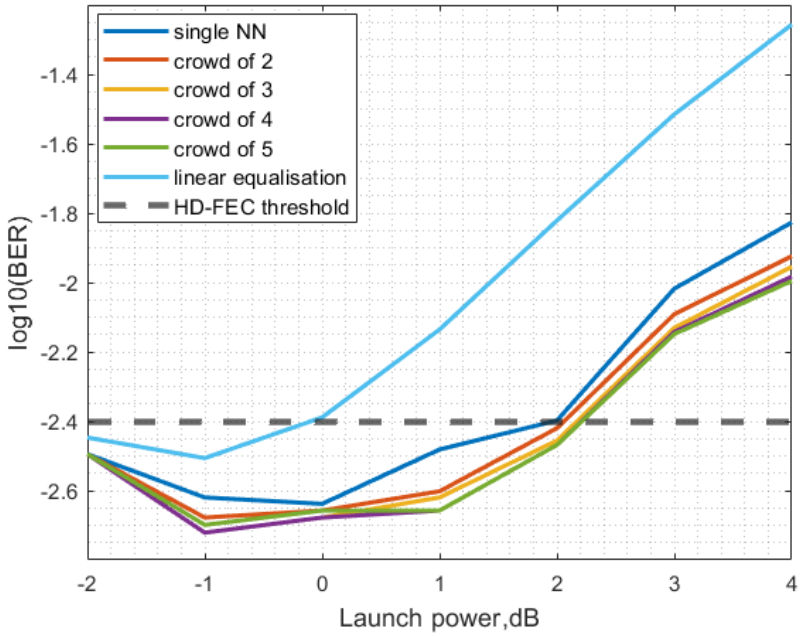

Figure 7.9: Resulted BER vs launch power at 60 dB SNR for the cases of the single NN vs combined (crowd) NNs, consisting of the same NNs when total CC increases with each member of crowd

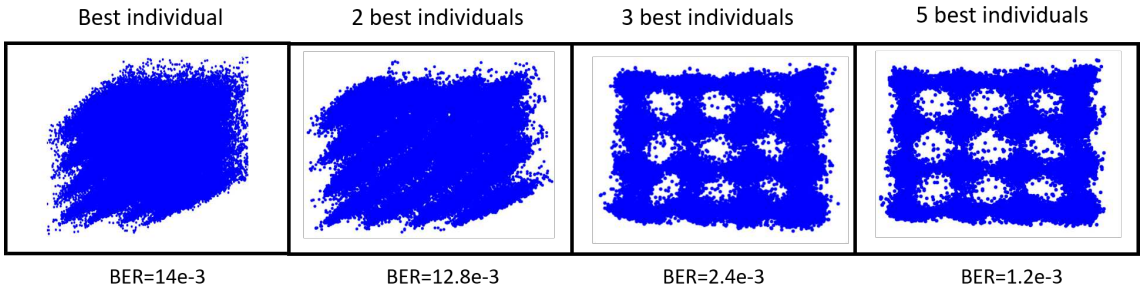

Figure 7.10: Improvement in BER and constellations for a crowd of 5 NNs when averaging takes place over 1, 2, 3, 4, 5 best individual NNs, which have the same configuration but different initialisation (overall CC is increasing).

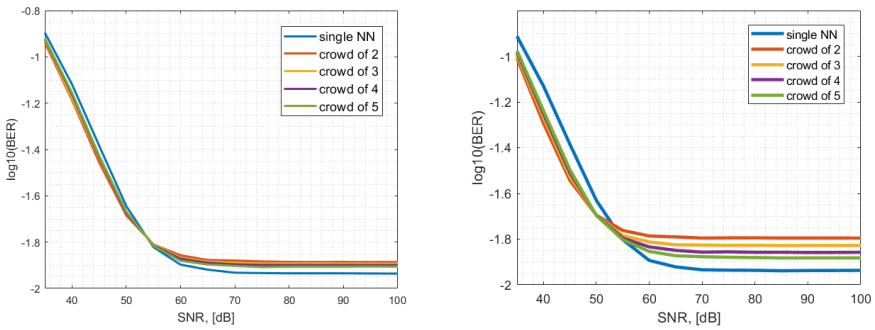

Figure 7.11: Resulted BER for the cases of the single NN vs combined (crowd) NNs, when SNR is between 35 and 100 dB at 4 dBm launch power when the crowd is combined of the same NNs captured at different values of epochs with the step size of a) 20 epochs; b)40 epochs

leading to distinct outputs. As the outputs of the NNs are nonlinear functions of these weights, they are likely to produce linearly independent outputs at different epochs.

Here, a step size of 20 epochs and 40 epochs have been examined. The structure of each NN model is the same as the one obtained in Table 4.1.

For instance, with a step size of 20 epochs, the model is stored at 400, 420, ..., up to 500 epochs. Similarly, for the crowd composed of the model captured at different epochs with a step size of 40, the snapshots of the model are taken at 300, 340, ..., and up to 500 epochs. Each stored model was then evaluated using the testing dataset to compare the corresponding values of the $BER_i(x^{test})$. The output of the crowd equalisation is subsequently determined using Eq. 7.1.

To test the performance of such an equaliser, the dataset from the 16 QAM 32-Gbaud transmission over 2000 km was used, described in section 4.2.2. The results of crowd equalisation with this particular structure are presented in Fig. 7.11.

Figure 7.11 demonstrates the results of this implementation of the crowd equalisation, displaying the BER performance as a function of SNR for various crowd sizes. Two different step sizes for epochs were examined, and the results indicate that using a step size of 40 yields superior improvement. Additionally, it becomes apparent that below an SNR of 55 dB, crowd equalisation results in only a small BER improvement. However, it can be seen that increasing the crowd size does not result in a noticeable performance improvement. Moreover, for higher SNR values, increasing the crowd size even leads to a distortion of the equalisation results.

These findings suggest that crowd equalisation, with different epoch instances, only offers marginal improvement at low SNR values. The limited benefit gained from increasing the crowd size may indicate that the epoch instances of the model are not as linearly independent as initially anticipated. It could also be because using this implementation of crowd equalisation makes it challenging to use the model's early-stopping feature, which could have resulted in a slight decline in the equaliser's performance due to overfitting. This prompts further investigation into the interdependencies and relationships between

the various epoch instances to gain a deeper understanding of their impact on crowd equalisation performance.

7.4 Conclusion

In this section, another solution to reduce the computational complexity of the equaliser was considered: an analogue implementation of NNs in the electrical or optical domain. The bottleneck of the optical implementation of NNs is the use of active devices to implement nonlinear activation nodes, which add noise to the computation chain and cause a cascade effect. In systems with more layers or neurons, this effect is more pronounced.

In order to address the issue of noise accumulation in deep neural network implementations, it was suggested to divide large networks into smaller ones and create "crowds" using these basic networks. This crowd equalisation is based on the principle of committee learning.

This chapter explored various implementations of crowd equalisation to address the challenge of noise propagation and amplification in neural network implementations. The investigation revealed that crowd implementation, which involves breaking down a single NN into smaller NNs, provides the most resilience against noise. However, it was also observed that at high SNR values, the performance of a single equaliser outperformed crowd equalisation with the same total computational complexity. This can be attributed to the larger NNs' ability to capture the signal evolution in the fibre more effectively.

Breaking down a single large NN into smaller ones requires a careful investigation into the design of the combination of the different structures of the NNs, as the solution is not so straightforward. Hence, simpler to-design implementations of crowd equalisation were examined, such as using identical NNs with different initial weights and neural networks captioned at different epochs.

In addition to the above, the testing of the other configurations of crowd equalisation has been conducted. However, the performance improvement was marginal in the cases of crowds at different learning rates. The potential structure to research is to attempt a crowd consisting of different machine learning algorithms. Still, it requires more timing and is more complex, as controlling the computational complexity of individual crowd members in that configuration is a more complicated task. Furthermore, it is essential to investigate the impact of utilising multiple NNs on power consumption.

Chapter 8

Conclusion and Further Work

8.1 Summary and conclusion

Kerr nonlinearity has been shown to be the most limiting factor in the performance of high-rate optical communication systems. Hence, to improve the achievable data rate and/or capacity, it is crucial to compensate for the undesired effects of these nonlinearities. Many of the existing techniques, such as DBP, IVSTF, NFT and OPC, have different types of drawbacks, where either they require high computational complexity, provide marginal performance benefit or need the knowledge of the optical parameters and difficult to reconfigure.

In recent years, machine learning has emerged as a promising approach for compensating fibre nonlinearities. However, complicated nonlinear interactions between noise and signal necessitate the use of a Machine learning equaliser with high computational complexity, making real-world implementation impossible. The training process for such equalisation would, in particular, necessitate a large amount of data and computational resources.

This thesis explores two ways to address this problem: designing low-complexity equalisers and simplifying the training process, which would result in a lower computational load and required electricity.

The design of low-complexity ML-based equalisers was done using intrinsically low CC solutions (like SVM & SVR), reducing the complexity of the achieved NNs (weight pruning), integrating optical solutions (OPC & DM), and considering photonic implementation of ML-based equalisers. The impact of these approaches on BER and computational complexity was evaluated.

Initially, SVM and SVR were investigated as potential methods for compensating fibre nonlinearities. These algorithms were chosen for their ability to handle complex decision boundaries effectively and their reputation as low-complexity approaches. SVM, a binary classification method, requires employing from $N-1$ SVMs to $N\times(N-1)$ SVMs, depending on the multi-class strategy used, to equalise an NQAM signal. On the other hand, SVR is a regression-based algorithm derived from SVM. Since SVR produces real-valued outputs, at least two SVR models would be needed to predict the real and imaginary

parts of the symbol for nonlinear equalisation. The system's performance was evaluated in the context of a 16QAM, 9 GBaud dispersion-managed optical transmission link. By using Bayesian optimisation, optimal equaliser parameters were determined. However, the results indicated only marginal improvements in system performance. The best BER enhancement was achieved with SVMs employing the "One vs One" multi-class strategy. Assessing the computational complexity required for the equaliser, it was determined that mitigating fibre nonlinearities in optical communication beyond the optimal launch power would necessitate a high number of multiplications. Consequently, SVM and SVR-based nonlinear equalisers were deemed impractical for real-time implementations as they offer only marginal performance improvements at a high computational cost.

Next, the focus shifted to a Neural Networks-based equalisation approach, specifically using a Complex-valued Multilayer Perceptron (MLP). MLP was chosen for its simple structure, relatively low computational complexity, and suitability for implementation using optical components. Two different implementations employing different structures and library packages were investigated. Similarly, a Bayesian optimisation was implemented to find the optimal structure of the equalisers for both implementations. Performance evaluation was conducted for 32 Gbaud 16QAM transmission over a distance of 2000 km. From there, it was found that the Keras-based implementation provides a better BER improvement for the system. In terms of computational complexity, the resulting equaliser required a high number of complex-valued multiplications. However, the NNs tend to get over-parameterised, prompting the implementation of weight-pruning techniques to simplify the models without significantly compromising their performance. The findings indicated that up to 60% of weights could be pruned at the optimum power while keeping the system performance below the HD-FEC threshold. Nevertheless, the resulting complexity remained relatively high, leading to the exploration of additional methods to further simplify the complexity of Neural Network-based equalisers.

In order to address the computational complexity of ML-based equalisers, an approach was explored that involved integrating optical solutions such as Optical Phase Conjugation (OPC) and Dispersion Managed (DM) links. The fundamental concept behind this approach is that optical solutions can effectively mitigate fibre distortions, reducing channel memory and minimising crosstalk between channels. By reducing channel memory, the number of consecutive symbols required as input for the Neural Networks (NNs) in the equaliser is impacted, subsequently influencing the processing delay and required computational complexity.

Firstly, the potential of OPC in reducing complexity in NN-based equalisers was investigated. The performance of equalisers employing only OPC, only NN, and OPC combined with NNs was compared for a 28-Gbaud 64-QAM transmission over a distance of 400 km. The results demonstrated that integrating OPC into NNs yielded additional improvements in BER performance. Furthermore, the number of complex-valued multiplications required for the equaliser was reduced. This investigation highlights the benefits of incorporating

optical solutions such as OPC into NN-based equalisers, leading to improved performance and decreased computational complexity. By leveraging the advantages of OPC and NNs together, it becomes possible to enhance the efficiency and effectiveness of equalisation in optical communication systems.

Similarly, the potential of Dispersion Managed (DM) links to reduce computational complexity was explored. One key distinction is that DM links require more amplifiers compared to unmanaged links, which in turn leads to increased nonlinearity. Consequently, when solely employing linear equalisation techniques, the performance of dispersion-managed links is typically inferior to that of uncompensated links. The performance of the equaliser was evaluated for PDM 32 Gbaud 16QAM fibre-optic communication over a 1000 km transmission link, considering both dispersion-compensated and uncompensated scenarios. Through Bayesian optimisation, an optimised configuration of Neural Networks (NNs) for both types of links was determined. Comparing the resulting BER performance of the NN- equalisers for both links, it was found that they were equivalent. However, when evaluating the resulting computational complexities, it was discovered that the DM link implementation required only 12% of the complex-valued multiplications compared to the uncompensated link implementation.

Furthermore, various techniques were explored to accelerate the learning process of NNs and enhance their effectiveness. One such technique involved data pruning, which aimed to remove redundant or repeated data. By leveraging the symmetry of the constellations in Quadrature Amplitude Modulation, a data folding approach was applied to prune the data. Although this technique significantly accelerated the learning process, it also led to a notable reduction in BER performance due to the loss of information on the symbols that overlap the fold boundaries.

Additionally, the concept of curriculum learning was investigated, which involves gradually shifting the training data from "easy" to more challenging instances during the learning process. This approach aimed to facilitate better learning outcomes by initially focusing on simpler examples and progressively introducing more complex scenarios. Similarly, it has shown the potential to speed up the learning.

Furthermore, the potential of a multi-output neural network was explored to facilitate the learning process of the NN-based equalisers. Instead of detecting a single symbol, the network was designed to detect three symbols simultaneously. It has shown its ability to not only enhance learning but also improve the BER performance of equalisation.

In addition, an investigation was conducted into a crowd equalisation approach based on committee learning, aiming to address noise accumulation in the photonic implementation of Neural Networks (NNs). Several methods of combining NNs for crowd equalisation were explored, including using different NN structures, the same structure with different initialisations, and capturing the same model at different epochs. For most implementations, the crowd members were selected to maintain the same computational complexity as a single optimised NN-based equaliser. The findings revealed that breaking down the

single large NNs into smaller ones increases the resilience to the noise. This can be attributed to the fact that linearly uncorrelated NNs can combine different perspectives, contributing to more accurate and robust predictions. Moreover, when implementing a crowd with the same structure but different initialisation while increasing the resulting computational complexity with each additional NN, it was possible to achieve results surpassing those obtained through Bayesian optimisation. However, it is crucial to consider power consumption when employing this crowd equalisation. While this approach shows promise in improving prediction accuracy and robustness, the increased computational complexity resulting from combining multiple NNs must be carefully balanced against the power requirements of the system.

8.2 Future work

In many of the experiments of this thesis, the focus has primarily been on a single-channel communication system. However, real-world optical networks often consist of multiple Wavelength Division Multiplexing (WDM) or Dense Wavelength Division Multiplexing (DWDM) channels. In a WDM or DWDM system, each channel may experience different impairments due to various factors such as fibre nonlinearities, chromatic dispersion, and polarisation effects. Nonlinear equalisers need to compensate for these impairments on a per-channel basis.

Compared to single-channel systems, WDM systems face more nonlinear challenges due to cross-channel interactions, such as XPM and FWM [15]. The challenges in WDM systems get more complicated as the number of channels increases and/or channel spacing decreases [111]. In order to effectively mitigate the impact of these challenges on each channel, sophisticated equalisation methods are required. In WDM systems, where multiple channels share a common physical medium like fibre, channel interaction or crosstalk becomes a significant concern. The presence of numerous channels increases complexity, as the equaliser must distinguish and compensate for each channel's unique characteristics while considering neighbouring channels' effects.

A conventional DBP can only account for intra-channel effects by back-propagating a single wavelength channel [24]. While the concept of multi-channel DBP has been suggested, its current hardware limitations make it impracticable as it requires substantial computational resources to handle inter-channel distortions in a WDM system [20].

Regarding Ml-equaliser, the fundamental principles of ML equalisation apply to both single-channel and WDM scenarios. However, as machine learning (ML) algorithms need to consider interactions between adjacent channels, the complexity and considerations involved (i.e. training data) expand significantly in a multi-channel context. Nonetheless, ML-based equalisation has proven to be effective in dealing with nonlinear data. This is especially true in the context of WDM channels, as evidenced by references [62; 69; 112; 113] demonstrating NN-based equalisers' potential to mitigate signal distortions arising

both from SPM and XPM for scenarios involving multiple channels. This paper [112], in particular, demonstrated that MLP can outperform standard DSP and conventional DBP (3StPS) in WDM systems.

However, it is still necessary to investigate the benefits of applying the low-complexity machine learning techniques described in this thesis for the WDM systems. Specifically, to assess the trade-off between reducing ML complexity and achieving BER improvement in WDM systems over traditional equalisation techniques. Specifically, the number of computations required to process the data from multiple channels might be too high, making it impractical for real-time implementation.

In addition, the findings and research presented in this thesis can be further developed in the following directions:

- Exploring the integration of optical solutions like Optical Phase Conjugation (OPC) and Dispersion Management (DM) with Neural Networks in Wavelength Division Multiplexing (WDM) systems.
- Demonstrating the influence of dispersion management on the complexity of ML-based equalisers through experimental data.
- Investigating the relationship between dispersion management and adaptability of NNs for different lengths of the fibre link and understanding how dispersion management affects adaptability and reconfiguration. Requirements of NN-based equalisation in optical communication systems can inform the development of more flexible and robust solutions.
- Examining the consequence of crowd equalisation on power consumption and exploring possible energy-efficient strategies for implementing crowd equalisation in optical communication.

By pursuing these directions, further advancements can be made to enhance the efficiency, performance, and adaptability of machine learning-based equalisation techniques in optical communication systems.

References

[1] M. Kamalian-Kopae, A. A. Ali, K. Nurlybayeva, A. Ellis, and S. Turitsyn, "Neural network-enhanced optical phase conjugation for nonlinearity mitigation," in *2022 Optical Fiber Communications Conference and Exhibition (OFC)*, pp. 1–3, 2022. 11, 71, 72, 76

[2] B. Mukherjee, *Optical WDM networks*. Springer Science & Business Media, 2006. 18

[3] D. Rafique and L. Velasco, "Machine learning for network automation: overview, architecture, and applications [invited tutorial]," *Journal of Optical Communications and Networking*, vol. 10, no. 10, pp. D126–D143, 2018. 18, 32, 33, 36, 37

[4] D. Zibar, M. Piels, R. Jones, and C. G. Schäeffer, "Machine learning techniques in optical communication," *Journal of Lightwave Technology*, vol. 34, no. 6, pp. 1442–1452, 2015. 18, 33, 36, 51

[5] E. Giacoumidis, Y. Lin, J. Wei, I. Aldaya, A. Tsokanos, and L. P. Barry, "Harnessing machine learning for fiber-induced nonlinearity mitigation in long-haul coherent optical ofdm," *Future internet*, vol. 11, no. 1, p. 2, 2018. 18, 35, 37

[6] F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of machine learning techniques in optical networks," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1383–1408, 2018. 18, 32, 33, 35, 37

[7] F. Musumeci, C. Rottondi, G. Corani, S. Shahkarami, F. Cugini, and M. Tornatore, "A tutorial on machine learning for failure management in optical networks," *Journal of Lightwave Technology*, vol. 37, no. 16, pp. 4125–4139, 2019. 18

[8] P. J. Freire, A. Napoli, B. Spinnler, N. Costa, S. K. Turitsyn, and J. E. Prilepsky, "Neural networks-based equalizers for coherent optical transmission: Caveats and pitfalls," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 28, no. 4, pp. 1–23, 2022. 18, 47, 51, 60

[9] G. P. Agrawal, *Fiber-optic communication systems*. John Wiley & Sons, 2012. 21, 28, 78

[10] N. Massa, "Fiber optic telecommunication," *Fundamental of Photonic, University of Connecticut*, 2000. 21, 22

[11] G. P. Agrawal, "Nonlinear fiber optics," in *Nonlinear Science at the Dawn of the 21st Century*, pp. 195–211, Springer, 2000. 23, 24, 25, 26

[12] P. J. Freire, A. Napoli, D. A. Ron, B. Spinnler, M. Anderson, W. Schairer, T. Bex, N. Costa, S. K. Turitsyn, and J. E. Prilepsky, "Reducing computational complexity of neural networks in optical channel equalization: From concepts to implementation," *Journal of Lightwave Technology*, pp. 1–26, 2023. 23

[13] A. Amari, X. Lin, O. A. Dobre, R. Venkatesan, and A. Alvarado, "A machine learning-based detection technique for optical fiber nonlinearity mitigation," *IEEE Photonics Technology Letters*, vol. 31, no. 8, pp. 627–630, 2019. 24

[14] A. E. Willner, S. M. Nezam, L. Yan, Z. Pan, and M. C. Hauer, "Monitoring and control of polarization-related impairments in optical fiber systems," *Journal of lightwave technology*, vol. 22, no. 1, pp. 106–125, 2004. 24

[15] R.-J. Essiambre, G. Kramer, P. J. Winzer, G. J. Foschini, and B. Goebel, "Capacity limits of optical fiber networks," *Journal of Lightwave Technology*, vol. 28, no. 4, pp. 662–701, 2010. 25, 115

[16] R. Dar and P. J. Winzer, "Nonlinear interference mitigation: Methods and potential gain," *Journal of Lightwave Technology*, vol. 35, no. 4, pp. 903–930, 2017. 26

[17] M. S. Faruk and S. J. Savory, "Digital signal processing for coherent transceivers employing multilevel formats," *Journal of Lightwave Technology*, vol. 35, no. 5, pp. 1125–1141, 2017. 28

[18] J. G. Proakis, *Digital communications*. McGraw-Hill, Higher Education, 2008. 28

[19] L. Szczecinski and A. Alvarado, *Bit-interleaved coded modulation: fundamentals, analysis and design*. John Wiley & Sons, 2015. 29

[20] J. C. Cartledge, F. P. Guiomar, F. R. Kschischang, G. Liga, and M. P. Yankov, "Digital signal processing for fiber nonlinearities," *Optics express*, vol. 25, no. 3, pp. 1916–1936, 2017. 30, 115

[21] E. Ip and J. M. Kahn, "Compensation of dispersion and nonlinear impairments using digital backpropagation," *Journal of Lightwave Technology*, vol. 26, no. 20, pp. 3416–3425, 2008. 30, 31

[22] D. Rafique, J. Zhao, and A. D. Ellis, "Digital back-propagation for spectrally efficient wdm 112 gbit/s pm m-ary qam transmission," *Optics Express*, vol. 19, no. 6, pp. 5219–5224, 2011. 30, 31

[23] G. Gao, J. Zhang, and W. Gu, "Analytical evaluation of practical dbp-based intra-channel nonlinearity compensators," *IEEE Photonics Technology Letters*, vol. 25, no. 8, pp. 717–720, 2013. 30, 31

[24] R. Dar and P. J. Winzer, "On the limits of digital back-propagation in fully loaded wdm systems," *IEEE Photonics Technology Letters*, vol. 28, no. 11, pp. 1253–1256, 2016. 30, 31, 37, 115

[25] C. Ju, N. Liu, X. Chen, and Z. Zhang, "Ssbi mitigation in a-rf-tone-based vssb-ofdm system with a frequency-domain volterra series equalizer," *Journal of Lightwave Technology*, vol. 33, no. 23, pp. 4997–5006, 2015. 30, 31

[26] E. Giacoumidis, I. Aldaya, M. A. Jarajreh, A. Tsokanos, S. T. Le, F. Farjady, Y. Jaouën, A. D. Ellis, and N. J. Doran, "Volterra-based reconfigurable nonlinear equalizer for coherent ofdm," *IEEE Photonics Technology Letters*, vol. 26, no. 14, pp. 1383–1386, 2014. 30, 31

[27] S. K. Turitsyn, J. E. Prilepsky, S. T. Le, S. Wahls, L. L. Frumin, M. Kamalian, and S. A. Derevyanko, "Nonlinear fourier transform for optical data processing and transmission: advances and perspectives," *Optica*, vol. 4, no. 3, pp. 307–322, 2017. 30, 31

[28] M. Kamalian-Kopae, A. Vasylchenkova, D. Shepelsky, J. E. Prilepsky, and S. K. Turitsyn, "Full-spectrum periodic nonlinear fourier transform optical communication through solving the riemann-hilbert problem," *Journal of Lightwave Technology*, vol. 38, no. 14, pp. 3602–3615, 2020. 30, 31

[29] A. Yariv, D. Fekete, and D. M. Pepper, "Compensation for channel dispersion by nonlinear optical phase conjugation," *Optics Letters*, vol. 4, no. 2, pp. 52–54, 1979. 30, 31, 69

[30] A. Chowdhury and R.-J. Essiambre, "Optical phase conjugation and pseudolinear transmission," *Optics letters*, vol. 29, no. 10, pp. 1105–1107, 2004. 30, 31

[31] A. D. Ellis, M. Tan, M. A. Iqbal, M. A. Z. Al-Khateeb, V. Gordienko, G. S. Mondaca, S. Fabbri, M. F. Stephens, M. E. McCarthy, A. Perentos, *et al.*, "4 tb/s transmission reach enhancement using 10× 400 gb/s super-channels and polarization insensitive dual band optical phase conjugation," *Journal of lightwave technology*, vol. 34, no. 8, pp. 1717–1723, 2016. 30, 31

[32] A. Ellis, M. McCarthy, M. Al-Khateeb, and S. Sygletos, "Capacity limits of systems employing multiple optical phase conjugators," *Optics express*, vol. 23, no. 16, pp. 20381–20393, 2015. 30, 31

[33] X. Liu, A. Chraplyvy, P. Winzer, R. Tkach, and S. Chandrasekhar, "Phase-conjugated twin waves for communication beyond the kerr nonlinearity limit," *Nature Photonics*, vol. 7, no. 7, pp. 560–568, 2013. 30, 31

[34] Y. Yu and J. Zhao, "Modified phase-conjugate twin wave schemes for fiber nonlinearity mitigation," *Optics Express*, vol. 23, no. 23, pp. 30399–30413, 2015. 30, 31

[35] T. Yoshida, T. Sugihara, K. Ishida, and T. Mizuochi, "Spectrally-efficient dual phase-conjugate twin waves with orthogonally multiplexed quadrature pulse-shaped signals," in *Optical Fiber Communication Conference*, pp. M3C–6, Optica Publishing Group, 2014. 30, 31

[36] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*, vol. 4. Springer, 2006. 32, 33, 53, 62, 87

[37] R. O. Duda, P. E. Hart, *et al.*, *Pattern classification and scene analysis*, vol. 3. Wiley New York, 1973. 32

[38] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012. 33

[39] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022. 33

[40] J. Mata, I. de Miguel, R. J. Duran, N. Merayo, S. K. Singh, A. Jukan, and M. Chamania, "Artificial intelligence (ai) methods in optical networks: A comprehensive survey," *Optical switching and networking*, vol. 28, pp. 43–57, 2018. 34, 37

[41] Huawei, "White paper on technological developments of optical networks." `https://https://www.huawei.com/en/huaweitech/industry-insights/technology/white‘papers/1/white_paper_on_optical_networks`. 35

[42] C. Rottondi, L. Barletta, A. Giusti, and M. Tornatore, "Machine-learning method for quality of transmission prediction of unestablished lightpaths," *Journal of Optical Communications and Networking*, vol. 10, no. 2, pp. A286–A297, 2018. 35, 36

[43] S. Aladin and C. Tremblay, "Cognitive tool for estimating the qot of new lightpaths," in *2018 Optical Fiber Communications Conference and Exposition (OFC)*, pp. 1–3, IEEE, 2018. 36

[44] E. Seve, J. Pesic, C. Delezoide, S. Bigo, and Y. Pointurier, "Learning process for reducing uncertainties on network parameters and design margins," *Journal of Optical Communications and Networking*, vol. 10, no. 2, pp. A298–A306, 2018. 36

[45] A. Caballero, J. C. Aguado, R. Borkowski, S. Saldaña, T. Jiménez, I. de Miguel, V. Arlunno, R. J. Durán, D. Zibar, J. B. Jensen, *et al.*, "Experimental demonstration

of a cognitive quality of transmission estimator for optical communication systems," *Optics express*, vol. 20, no. 26, pp. B64–B70, 2012. 36

[46] E. d. A. Barboza, C. J. Bastos-Filho, J. F. Martins-Filho, U. C. de Moura, and J. R. de Oliveira, "Self-adaptive erbium-doped fiber amplifiers using machine learning," in *2013 SBMO/IEEE MTT-S International Microwave & Optoelectronics Conference (IMOC)*, pp. 1–5, IEEE, 2013. 36

[47] Y. Huang, C. L. Gutterman, P. Samadi, P. B. Cho, W. Samoud, C. Ware, M. Lourdiane, G. Zussman, and K. Bergman, "Dynamic mitigation of edfa power excursions with machine learning," *Optics express*, vol. 25, no. 3, pp. 2245–2258, 2017. 36

[48] M. C. Tan, F. N. Khan, W. H. Al-Arashi, Y. Zhou, and A. P. T. Lau, "Simultaneous optical performance monitoring and modulation format/bit-rate identification using principal component analysis," *Journal of Optical Communications and Networking*, vol. 6, no. 5, pp. 441–448, 2014. 36

[49] F. N. Khan, Y. Zhou, A. P. T. Lau, and C. Lu, "Modulation format identification in heterogeneous fiber-optic networks using artificial neural networks," *Optics express*, vol. 20, no. 11, pp. 12422–12431, 2012. 36

[50] J. Thrane, J. Wass, M. Piels, J. C. Diniz, R. Jones, and D. Zibar, "Machine learning techniques for optical performance monitoring from directly detected pdm-qam signals," *Journal of Lightwave Technology*, vol. 35, no. 4, pp. 868–875, 2016. 36

[51] N. G. Gonzalez, D. Zibar, and I. T. Monroy, "Cognitive digital receiver for burst mode phase modulated radio over fiber links," in *36th European Conference and Exhibition on Optical Communication*, pp. 1–3, IEEE, 2010. 36

[52] M. A. Jarajreh, E. Giacoumidis, I. Aldaya, S. T. Le, A. Tsokanos, Z. Ghassemlooy, and N. J. Doran, "Artificial neural network nonlinear equalizer for coherent optical ofdm," *IEEE Photonics Technology Letters*, vol. 27, no. 4, pp. 387–390, 2014. 35, 36, 58

[53] P. Li, L. Yi, L. Xue, and W. Hu, "56 gbps im/dd pon based on 10g-class optical devices with 29 db loss budget enabled by machine learning," in *2018 Optical Fiber Communications Conference and Exposition (OFC)*, pp. 1–3, IEEE, 2018. 36

[54] C.-Y. Chuang, L.-C. Liu, C.-C. Wei, J.-J. Liu, L. Henrickson, W.-J. Huang, C.-L. Wang, Y.-K. Chen, and J. Chen, "Convolutional neural network based nonlinear classifier for 112-gbps high speed optical link," in *Optical Fiber Communication Conference*, pp. W2A–43, Optica Publishing Group, 2018. 36

[55] C. Häger and H. D. Pfister, "Nonlinear interference mitigation via deep neural networks," in *Optical fiber communication conference*, pp. W3A–4, Optical Society of America, 2018. 36

[56] D. Wang, M. Zhang, M. Fu, Z. Cai, Z. Li, H. Han, Y. Cui, and B. Luo, "Nonlinearity mitigation using a machine learning detector based on k-nearest neighbors," *IEEE Photonics Technology Letters*, vol. 28, no. 19, pp. 2102–2105, 2016. 36

[57] E. Giacoumidis, S. Mhatli, M. F. Stephens, A. Tsokanos, J. Wei, M. E. McCarthy, N. J. Doran, and A. D. Ellis, "Reduction of nonlinear intersubcarrier intermixing in coherent optical ofdm by a fast newton-based support vector machine nonlinear equalizer," *Journal of Lightwave Technology*, vol. 35, no. 12, pp. 2391–2397, 2017. 36

[58] D. Wang, M. Zhang, Z. Li, Y. Cui, J. Liu, Y. Yang, and H. Wang, "Nonlinear decision boundary created by a machine learning-based classifier to mitigate nonlinear phase noise," in *2015 European Conference on Optical Communication (ECOC)*, pp. 1–3, IEEE, 2015. 36

[59] J. Zhang, W. Chen, M. Gao, B. Chen, and G. Shen, "Novel low-complexity fully-blind density-centroid–tracking equalizer for 64-qam coherent optical communication systems," in *Optical Fiber Communication Conference*, pp. M1G–4, Optical Society of America, 2018. 36

[60] T. Tanimura, T. Hoshida, J. C. Rasmussen, M. Suzuki, and H. Morikawa, "Osnr monitoring by deep neural networks trained with asynchronously sampled data," in *2016 21st OptoElectronics and Communications Conference (OECC) held jointly with 2016 International Conference on Photonics in Switching (PS)*, pp. 1–3, IEEE, 2016. 36

[61] F. Meng, S. Yan, K. Nikolovgenis, Y. Ou, R. Wang, Y. Bi, E. Hugues-Salas, R. Nejabati, and D. Simeonidou, "Field trial of gaussian process learning of function-agnostic channel performance under uncertainty," in *Optical Fiber Communication Conference*, pp. W4F–5, Optica Publishing Group, 2018. 36

[62] P. J. Freire, Y. Osadchuk, B. Spinnler, A. Napoli, W. Schairer, N. Costa, J. E. Prilepsky, and S. K. Turitsyn, "Performance versus complexity study of neural network equalizers in coherent optical systems," *Journal of Lightwave Technology*, vol. 39, no. 19, pp. 6085–6096, 2021. 35, 37, 51, 115

[63] F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of machine learning techniques in optical networks," *IEEE Communications Surveys Tutorials*, vol. 21, no. 2, pp. 1383–1408, 2019. 36, 47, 51

[64] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995. 38, 39, 42

[65] S. Hassan, N. Tariq, R. A. Naqvi, A. U. Rehman, and M. K. Kaabar, "Performance evaluation of machine learning-based channel equalization techniques: new trends and challenges," *Journal of Sensors*, vol. 2022, pp. 1–14, 2022. 38

[66] W. Chen, J. Zhang, M. Gao, and G. Shen, "Performance improvement of 64-qam coherent optical communication system by optimizing symbol decision boundary based on support vector machine," *Optics Communications*, vol. 410, pp. 1–7, 2018. 38, 39

[67] C. J. Burges and B. Schölkopf, "Improving the accuracy and speed of support vector machines," *Advances in neural information processing systems*, vol. 9, 1996. 39

[68] C. Wang, J. Du, G. Chen, H. Wang, L. Sun, K. Xu, B. Liu, and Z. He, "Qam classification methods by svm machine learning for improved optical interconnection," *Optics Communications*, vol. 444, pp. 1–8, 2019. 40

[69] O. Sidelnikov, A. Redyuk, and S. Sygletos, "Equalization performance and complexity analysis of dynamic deep neural networks in long haul transmission systems," *Optics Express*, vol. 26, no. 25, p. 32765, 2018. 51, 115

[70] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, "All-optical machine learning using diffractive deep neural networks," *Science*, vol. 361, no. 6406, pp. 1004–1008, 2018. 52

[71] Y. Zuo, B. Li, Y. Zhao, Y. Jiang, Y.-C. Chen, P. Chen, G.-B. Jo, J. Liu, and S. Du, "All-optical neural network with nonlinear activation functions," *Optica*, vol. 6, no. 9, pp. 1132–1137, 2019. 52

[72] M. Miscuglio, A. Mehrabian, Z. Hu, S. I. Azzam, J. George, A. V. Kildishev, M. Pelton, and V. J. Sorger, "All-optical nonlinear activation function for photonic neural networks," *Opt. Mater. Express*, vol. 8, pp. 3851–3863, Dec 2018. 52

[73] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, "A survey on modern trainable activation functions," *Neural Networks*, vol. 138, pp. 14–32, 2021. 53

[74] B. Karlik and A. V. Olgac, "Performance analysis of various activation functions in generalized mlp architectures of neural networks," *International Journal of Artificial Intelligence and Expert Systems*, vol. 1, no. 4, pp. 111–122, 2011. 54

[75] J. R. Barry, *Wireless infrared communications*, vol. 280. Springer Science & Business Media, 1994. 54

[76] T. Kim and T. Adali, "Fully complex multi-layer perceptron network for nonlinear signal processing," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 32, pp. 29–43, 2002. 54

[77] M. Nakamura, Y. Fukumoto, S. Owaki, T. Sakamoto, and N. Yamamoto, "Experimental demonstration of spm compensation using a complex-valued neural network for 40-gbit/s optical 16qam signals," *IEICE Communications Express*, vol. 8, no. 8, pp. 281–286, 2019. 54

[78] T. A. Eriksson, H. Bülow, and A. Leven, "Applying neural networks in optical communication systems: Possible pitfalls," *IEEE Photonics Technology Letters*, vol. 29, no. 23, pp. 2091–2094, 2017. 56

[79] J. A. Barrachina, "Complex-valued neural networks (cvnn)," Jan. 2021. 58

[80] Y. Kuroe, M. Yoshid, and T. Mori, "On activation functions for complex-valued neural networks—existence of energy functions—," in *Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003*, pp. 985–992, Springer, 2003. 58

[81] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015. 58

[82] M. Schaedler, C. Bluemm, M. Kuschnerov, F. Pittalà, S. Calabrò, and S. Pachnicke, "Deep neural network equalization for optical short reach communication," *Applied Sciences*, vol. 9, no. 21, p. 4675, 2019. 58

[83] J. Zhang, P. Lei, S. Hu, M. Zhu, Z. Yu, B. Xu, and K. Qiu, "Functional-link neural network for nonlinear equalizer in coherent optical fiber communications," *IEEE Access*, vol. 7, pp. 149900–149907, 2019. 58

[84] O. Kotlyar, M. Kamalian-Kopae, M. Pankratova, A. Vasylchenkova, J. E. Prilepsky, and S. K. Turitsyn, "Convolutional long short-term memory neural network equalizer for nonlinear fourier transform-based optical transmission systems," *Optics Express*, vol. 29, no. 7, pp. 11254–11267, 2021. 58

[85] H. Ming, X. Chen, X. Fang, L. Zhang, C. Li, and F. Zhang, "Ultralow complexity long short-term memory network for fiber nonlinearity mitigation in coherent optical communication systems," *Journal of Lightwave Technology*, vol. 40, no. 8, pp. 2427–2434, 2022. 58

[86] T. Hoefler, D. Alistarh, T. Ben-Nun, N. Dryden, and A. Peste, "Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks," 2021. 64

[87] Z. Allen-Zhu, Y. Li, and Z. Song, "A convergence theory for deep learning via over-parameterization," in *International Conference on Machine Learning*, pp. 242–252, PMLR, 2019. 64

[88] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," 2016. 64

[89] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017. 64

[90] T. Liang, J. Glossner, L. Wang, S. Shi, and X. Zhang, "Pruning and quantization for deep neural network acceleration: A survey," 2021. 64

[91] Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, *et al.*, "Deep learning with coherent nanophotonic circuits," *Nature Photonics*, vol. 11, no. 7, pp. 441–446, 2017. 69, 99

[92] L. Wang, M. Gao, Y. Zhang, F. Cao, and H. Huang, "Optical phase conjugation with complex-valued deep neural network for wdm 64-qam coherent optical systems," *IEEE Photonics Journal*, 2021. 70

[93] G. Saavedra, G. Liga, and P. Bayvel, "Volterra-assisted optical phase conjugation: A hybrid optical-digital scheme for fiber nonlinearity compensation," *JLT*, vol. 37, no. 10, pp. 2467–2479, 2019. 70

[94] A. Ali, T. T. Nguyen, S. Boscolo, S. Takasaka, R. Sugizaki, and A. D. Ellis, "Reduced impact of frequency dithering on the performance of high-order modulation format phase conjugation," in *2021 OFC*, pp. 1–3, IEEE, 2021. 70, 71

[95] A. Napoli, Z. Maalej, V. A. Sleiffer, M. Kuschnerov, D. Rafique, E. Timmers, B. Spinnler, T. Rahman, L. D. Coelho, and N. Hanik, "Reduced complexity digital back-propagation methods for optical communication systems," *Journal of lightwave technology*, vol. 32, no. 7, pp. 1351–1362, 2014. 76

[96] S. Lawan and M. Ajiya, "Dispersion management in a single-mode optical fiber communication system using dispersion compensating fiber," in *2013 IEEE International Conference on Emerging & Sustainable Technologies for Power & ICT in a Developing Society (NIGERCON)*, pp. 93–95, IEEE, 2013. 77

[97] T. L. Singal, *Dispersion Management Techniques*, p. 279–318. Cambridge University Press, 2016. 78

[98] D. Breuer, F. Küppers, A. Mattheus, E. Shapiro, I. Gabitov, and S. Turitsyn, "Symmetrical dispersion compensation for standard monomode-fiber-based communication systems with large amplifier spacing," *Optics Letters*, vol. 22, no. 13, pp. 982–984, 1997. 78

[99] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," vol. 60, p. 6, 06 2009. 92

[100] P. Soviany, R. T. Ionescu, P. Rota, and N. Sebe, "Curriculum learning: A survey," *International Journal of Computer Vision*, vol. 130, no. 6, pp. 1526–1565, 2022. 92

[101] G. Wetzstein, A. Ozcan, S. Gigan, S. Fan, D. Englund, M. Soljačić, C. Denz, D. A. Miller, and D. Psaltis, "Inference in artificial intelligence with deep optics and photonics," *Nature*, vol. 588, no. 7836, pp. 39–47, 2020. 99

[102] R. Hamerly, L. Bernstein, A. Sludds, M. Soljačić, and D. Englund, "Large-scale optical neural networks based on photoelectric multiplication," *Physical Review X*, vol. 9, no. 2, p. 021032, 2019. 99

[103] B. J. Shastri, A. N. Tait, T. F. de Lima, M. A. Nahmias, H.-T. Peng, and P. R. Prucnal, "Principles of neuromorphic photonics," *arXiv preprint arXiv:1801.00016*, 2017. 99

[104] X. Xu, M. Tan, B. Corcoran, J. Wu, T. G. Nguyen, A. Boes, S. T. Chu, B. E. Little, R. Morandotti, A. Mitchell, *et al.*, "Photonic perceptron based on a kerr microcomb for high-speed, scalable, optical neural networks," *Laser & Photonics Reviews*, vol. 14, no. 10, p. 2000070, 2020. 99

[105] G. Mourgias-Alexandris, G. Dabos, N. Passalis, A. Totović, A. Tefas, and N. Pleros, "All-optical wdm recurrent neural networks with gating," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 26, no. 5, pp. 1–7, 2020. 99

[106] J. R. Ong, C. C. Ooi, T. Y. Ang, S. T. Lim, and C. E. Png, "Photonic convolutional neural networks using integrated diffractive optics," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 26, no. 5, pp. 1–8, 2020. 99

[107] N. Semenova, X. Porte, L. Andreoli, M. Jacquot, L. Larger, and D. Brunner, "Fundamental aspects of noise in analog-hardware neural networks," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 29, no. 10, p. 103128, 2019. 100, 102

[108] C. Zhou, P. Kadambi, M. Mattina, and P. N. Whatmough, "Noisy machines: Understanding noisy neural networks and enhancing robustness to analog hardware errors using distillation," *arXiv preprint arXiv:2001.04974*, 2020. 102

[109] N. Nagabushan, N. Satish, and S. Raghuram, "Effect of injected noise in deep neural networks," in *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1–5, IEEE, 2016. 102

[110] Z. He, A. S. Rakin, and D. Fan, "Parametric noise injection: Trainable randomness to improve deep neural network robustness against adversarial attack," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 588–597, 2019. 102

[111] A. Amari, O. A. Dobre, R. Venkatesan, O. S. Kumar, P. Ciblat, and Y. Jaouën, "A survey on fiber nonlinearity compensation for 400 gb/s and beyond optical communication systems," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 3097–3113, 2017. 115

[112] P. J. Freire, Y. Osadchuk, B. Spinnler, W. Schairer, A. Napoli, N. Costa, J. E. Prilepsky, and S. K. Turitsyn, "Experimental study of deep neural network equalizers performance in optical links," in *2021 Optical Fiber Communications Conference and Exhibition (OFC)*, pp. 1–3, 2021. 115, 116

[113] R. Jiang, Z. Fu, Y. Bao, H. Wang, X. Ding, and Z. Wang, "Data-driven method for nonlinear optical fiber channel modeling based on deep neural network," *IEEE Photonics Journal*, vol. 14, no. 4, pp. 1–8, 2022. 115

Appendix A

Appendix

```
Code modifications for the data folding and unfolding
def folding(data):
        data_1=real(data)+1j*|imag(data)|
        F1=imag(data)>=0
        data_2=|real(data_1)|+1j*imag(data_1)
        F2=real(data_1)>=0
        return [data_2, F1, F2]

def unfolding(data_unf, F1U, F2U):
        data_2_unf[F2U==0]=-real(data_unf[F2U==0])+1j*imag(data_unf[F2U==0])
        data_1_unf[F1U==0]=real(data_2_unf[F1U==0])-1j*imag(data_2_unf[F1U==0])
        return data_1_unf
```

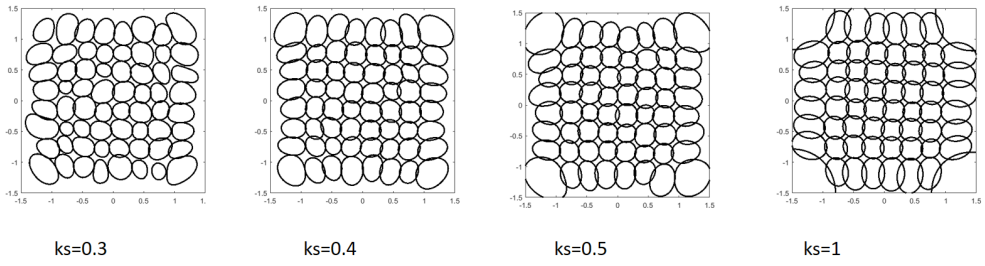

Figure A.1: The impact of the change in kernel scale on the shape of the decision boundary of SVM

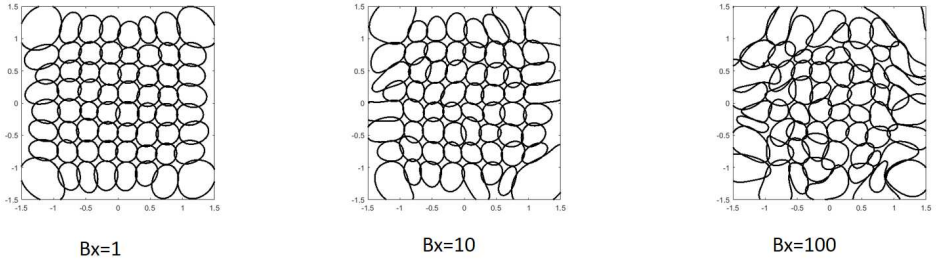

Figure A.2: The impact of the change in box constraint on the shape of the decision boundary of SVM

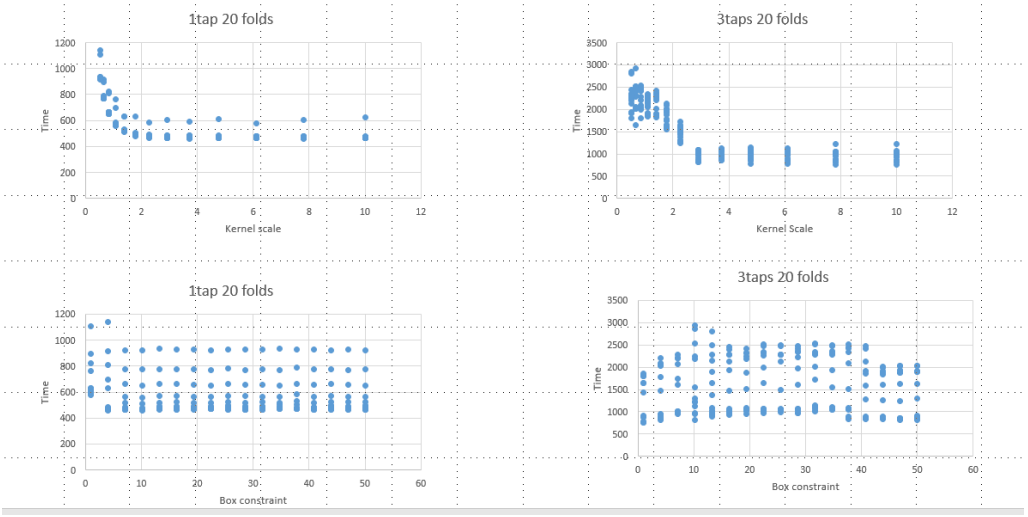

Figure A.3: The impact of the kernel scale and box constraints on the processing time of SVM-based equalisation

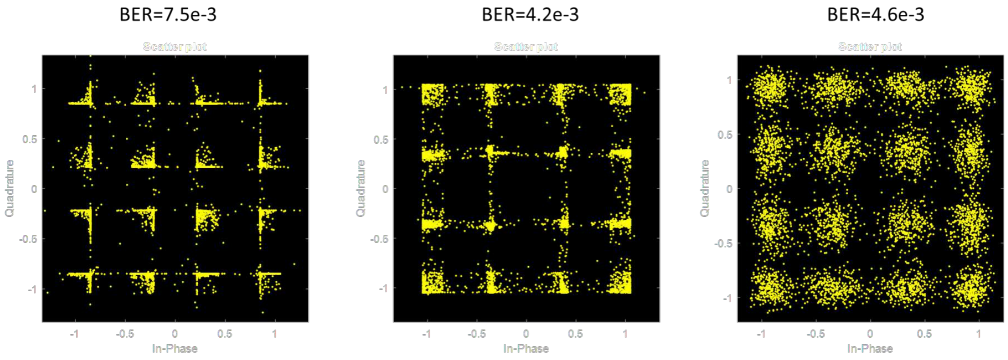

Figure A.4: The impact of the change in kernel scale on the constellations of SVR

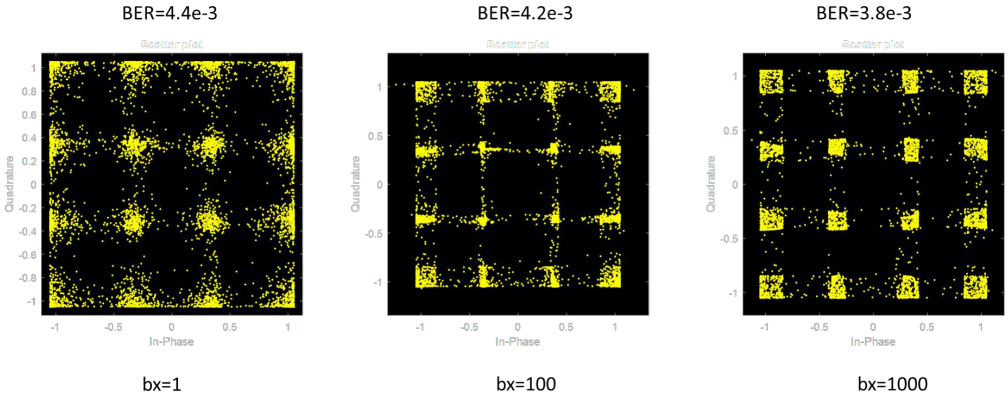

Figure A.5: The impact of the change in box constraint on the constellations of SVR

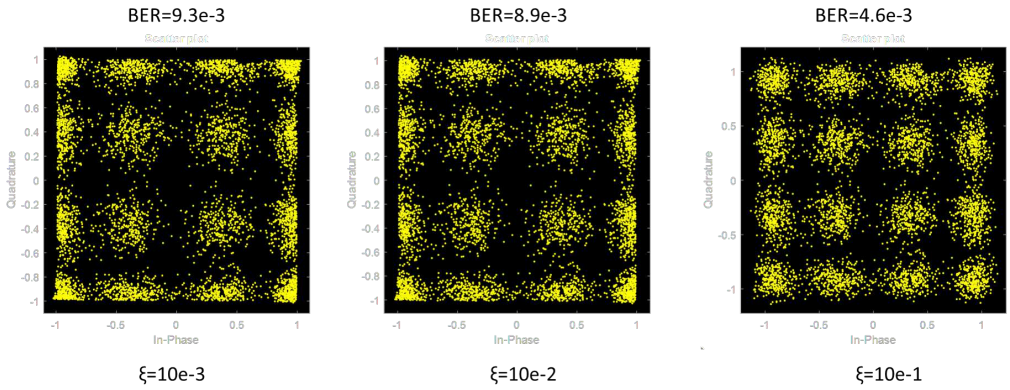

Figure A.6: The impact of the change in ξ on the constellations of SVR

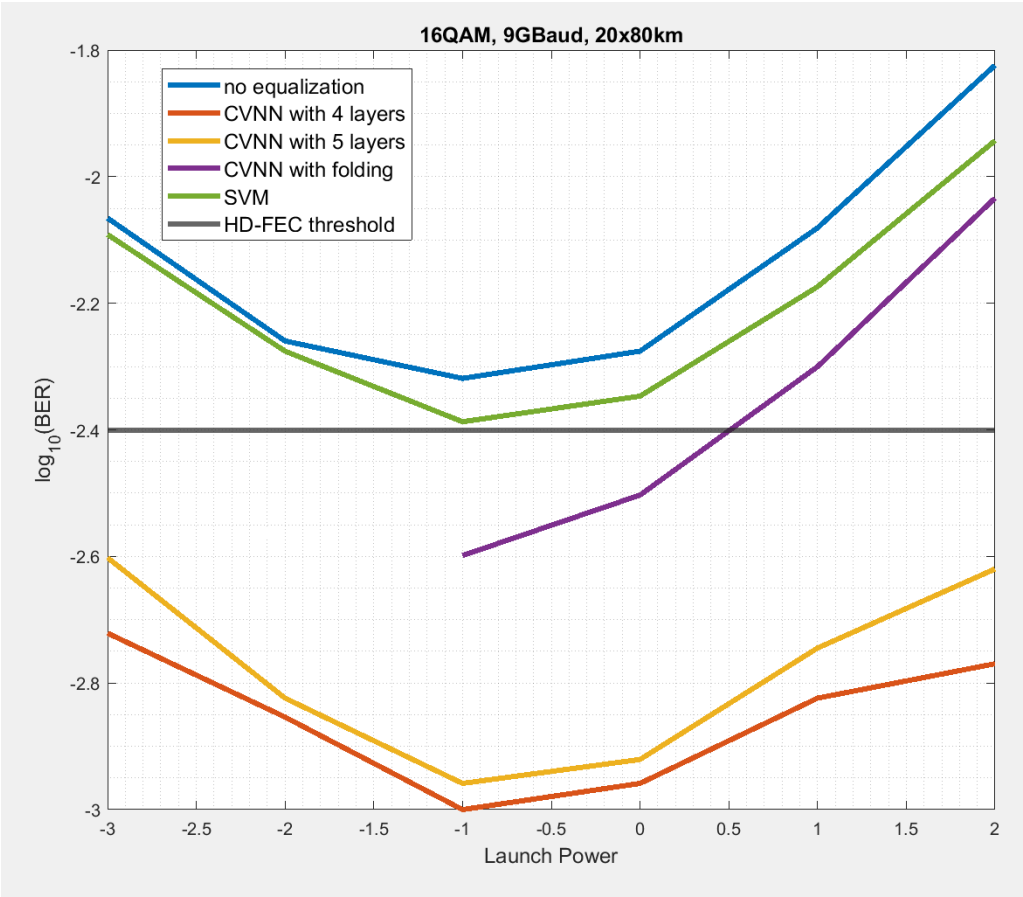

Figure A.7: The comparison of BER results vs launch power between SVM and basic implementation of CVNN, the data for is from 3.5

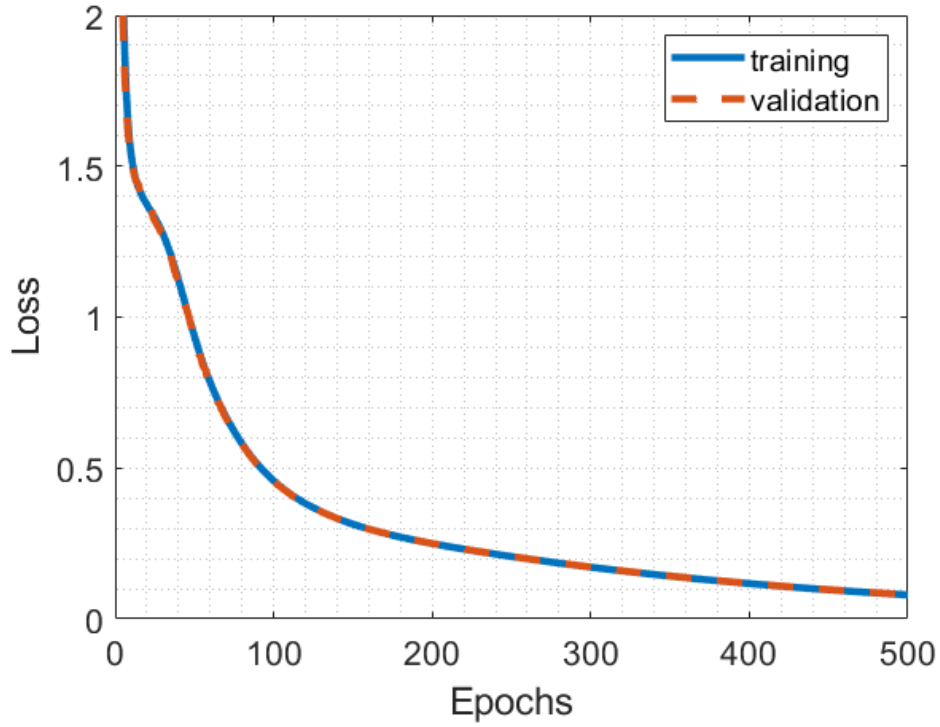

Figure A.8: Learning curves for CVNNs

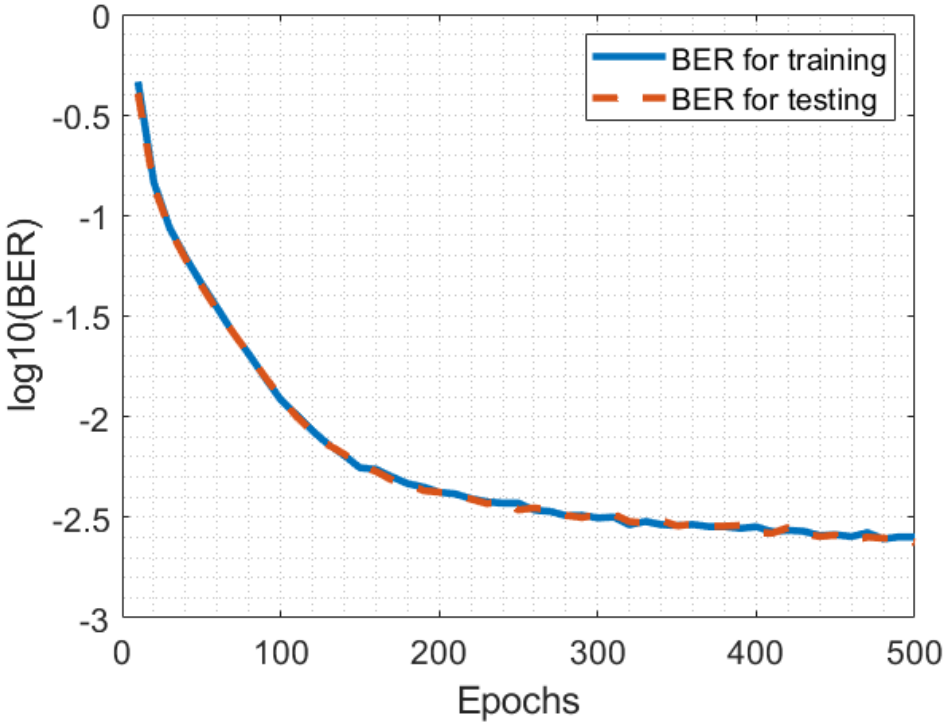

Figure A.9: BER improvement using CVNN as the number of epochs increases

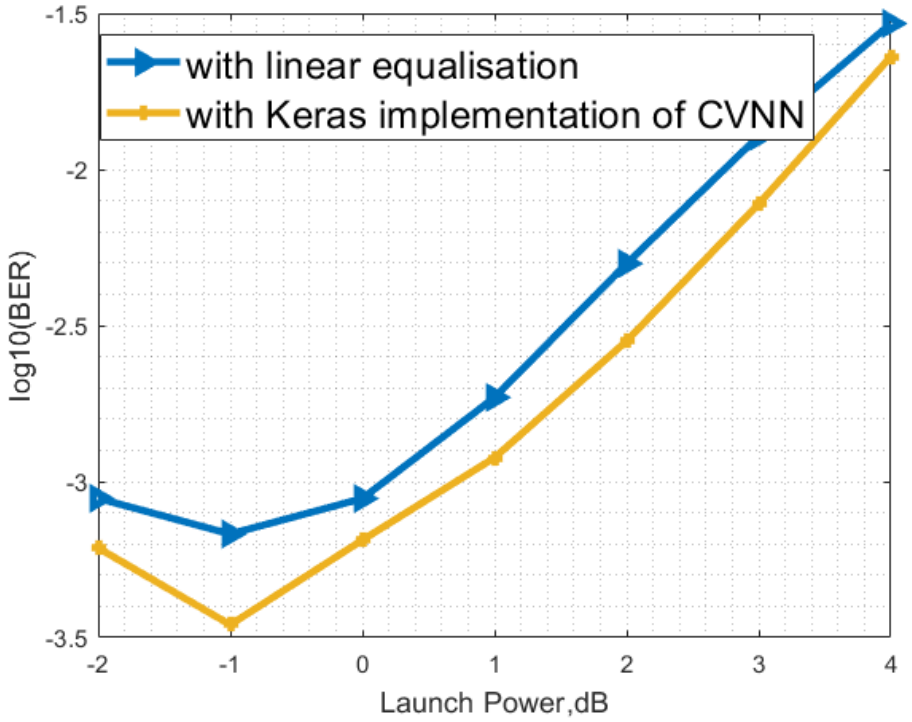

Figure A.10: BER vs signal power with the Keras implementation of Neural Network equaliser at the receiver when span length is 80 km

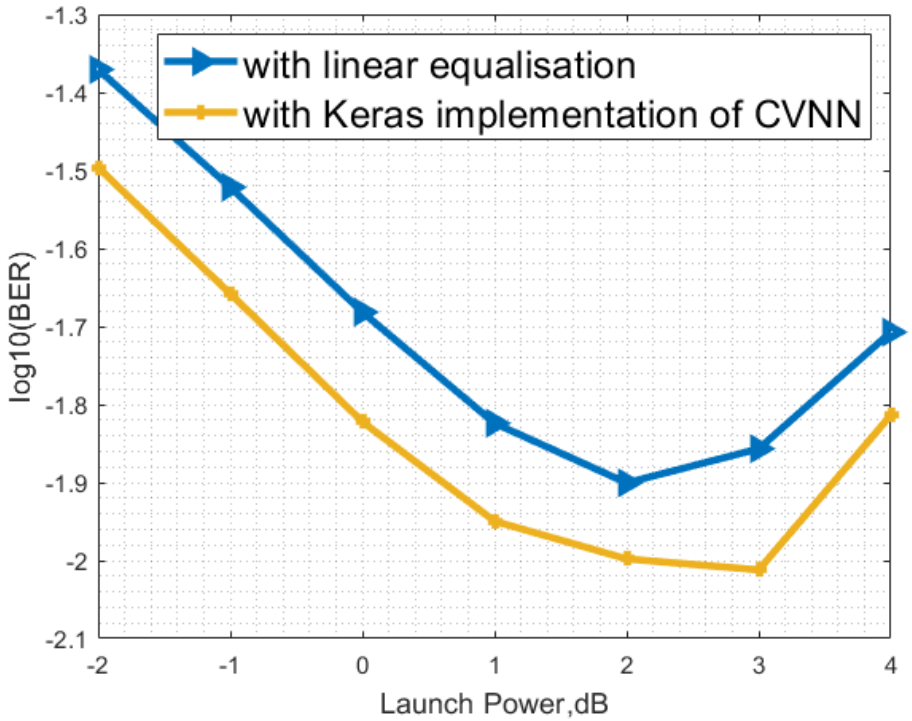

Figure A.11: BER vs signal power with the Keras implementation of Neural Network equaliser at the receiver when span length is 120 km

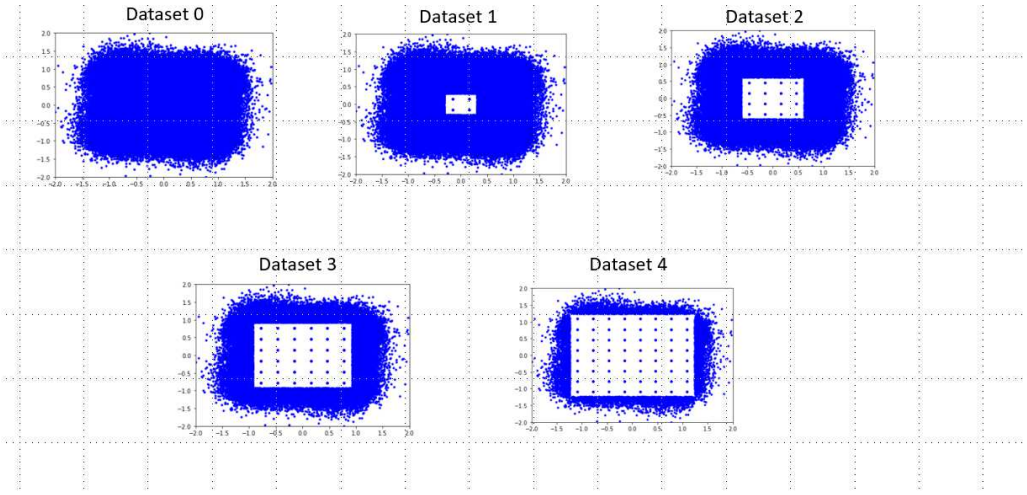

Figure A.12: Modification of the dataset for the curriculum learning from "easy to difficult for the 64 QAM system

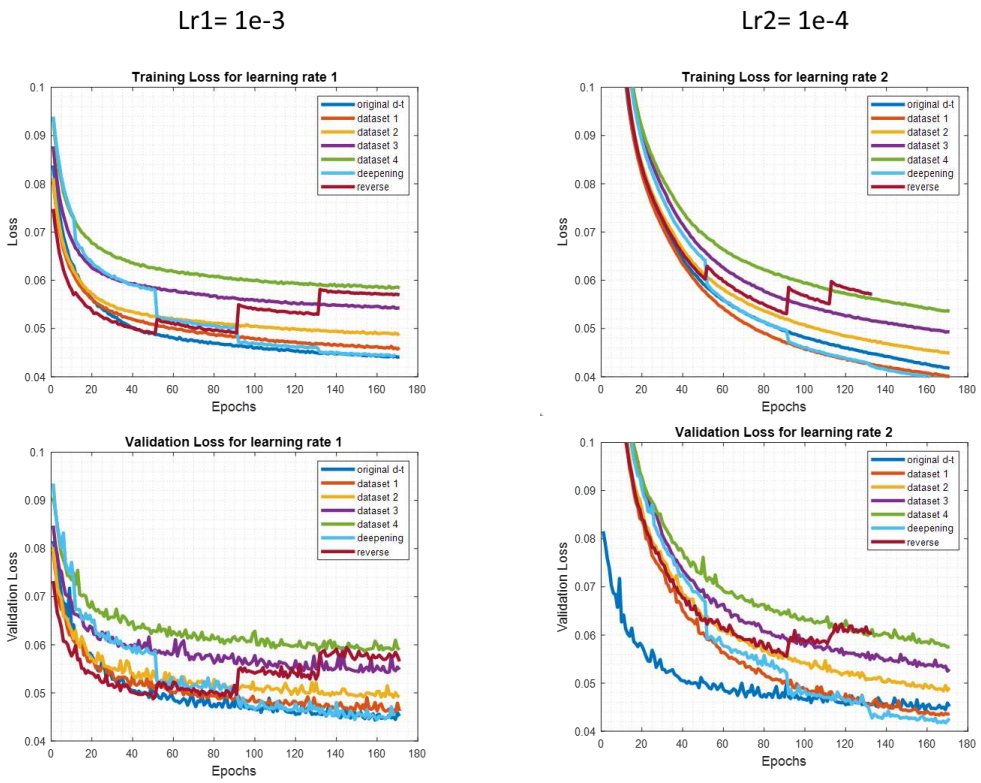

Figure A.13: Impact of the different datasets on the training and validation losses, when different datasets were applied, where the datasets are demonstrated above, "deepening" corresponds to a gradual increase in the dataset complexity (from dataset 4 to 0) and "reverse" corresponds to the opposite simplification (from dataset 0 to 4) in the dataset complexity.

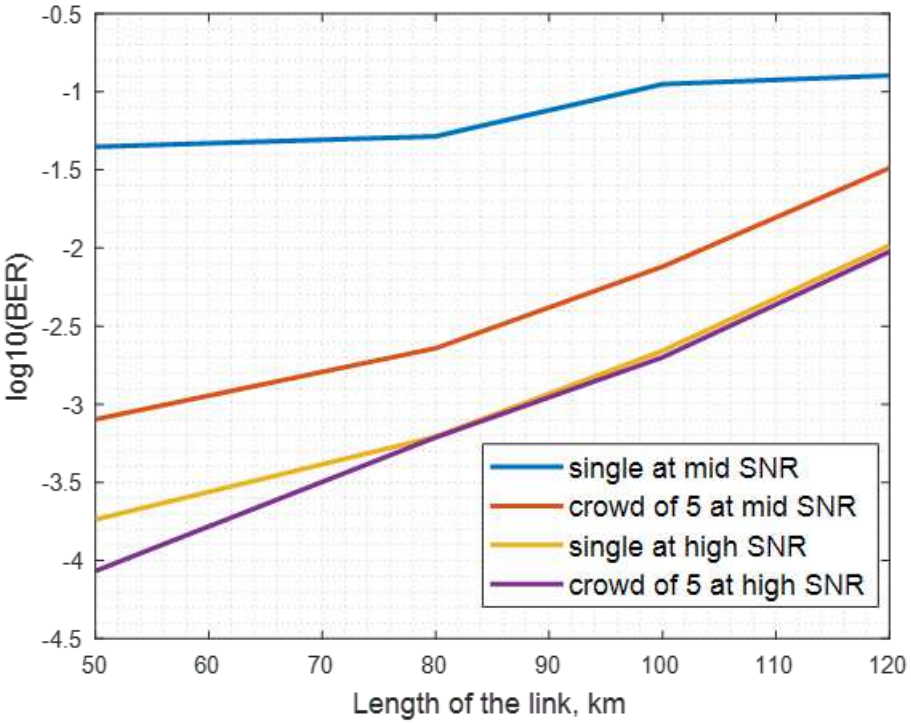

Figure A.14: Resulted BER at mid and high values of SNR for the cases of the single NN vs crowd NN (consisting of the same NNs when total CC increases with each member of the crowd) at the optimum power as a function of the span length.