LEARNED VOLTERRA MODELS FOR NONLINEARITY EQUALISATION IN OPTICAL NETWORKS

Nelson Castro Salgado

Doctor of Philosophy

ASTON UNIVERSITY

December 2024

Abstract

Learned Volterra Models For Nonlinearity Equalisation in Optical Networks
Nelson Castro Salgado
Doctor of Philosophy
2024

Digital equalisation of fibre-based nonlinear impairments in optical transmission systems remains commercially unavailable, mainly due to the high computational complexity of existing algorithms. Machine learning has recently revolutionised the field, enabling low-complexity schemes and introducing a range of approaches, from black-box methods to model-driven schemes. While black-box schemes are effective, they lack interpretability, require extensive training data and rely on heuristic designs. In contrast, model-driven schemes, such as learned digital backpropagation (LDBP), integrate signal propagation principles into the equaliser architecture, providing a framework that can be more easily understood and optimised. Although LDBP has achieved significant performance improvements and cost reductions, its sequential computations lead to high processing latency. For high-speed applications, the architecture of Volterra series models is an attractive alternative due to their inherent parallelisation capabilities. In particular, the third-order inverse Volterra series transfer function (IVSTF) model, while having known accuracy limitations that hinder its applicability, features a fully parallel structure with untapped potential as the basis of model-driven schemes.

This thesis presents a learned Volterra-based framework to mitigate nonlinear impairments, providing an alternative to LDBP. For single-channel transmission, we present a time-domain equaliser enabled by machine learning and based on simplifying the IVSTF. The scheme achieves equivalent performance to LDBP with comparable computational effort. For wavelength-division multiplexed (WDM) systems, three multiple-input-multiple-output (MIMO) equalisation architectures for mitigating interchannel impairments are introduced. Their design and training were enabled by a purpose-built computational framework. Efficient MIMO equalisation is achieved, which has not been demonstrated before with the IVSTF architecture. The proposed models demonstrate robust improvements over chromatic dispersion compensation. A comprehensive performance and cost analysis identifies the model with the best trade-off, and the interpretability of our approach is demonstrated through the examination of the learned parameters. Our analysis and results can be used as guidelines for designing learned multi-channel equalisers for WDM systems.

Keywords:
Digital Nonlinearity Equalisation, Digital Signal Processing, Machine Learning, Volterra Series, Model-driven, Optical Fibre Networks

To my family, for their love and support.

Personal Acknowledgements

This work would not have been possible without the support of many people. I am deeply grateful to my main supervisor, Stylianos Sygletos, for his guidance and encouragement throughout this journey. I was truly inspired by his passion for research and curiosity. His sharp eye helped me become more thorough and careful, and I learned greatly from his tireless work ethic.

I was fortunate to have an excellent supervisory team. I am thankful to my associate supervisors, Andrew Ellis and Sonia Boscolo. Sonia joined at a critical moment in my PhD, and her belief in my project and encouragement motivated me to persevere. Meetings with Andrew were always inspiring; his insights and timely advice were invaluable.

My thanks also extend to Sergei Turitsyn, the director of AIPT, who supported me financially during the challenging final months of my PhD.

I was very fortunate to undertake an internship at Microsoft at the end of my first year. I am especially grateful to Thomas Karagiannis, my internship supervisor, for allowing me to work with him and his team. I would also like to thank Shawn Siew, who generously shared his knowledge and insights with me during my time there.

Being a PhD student at AIPT offered many opportunities to meet amazing people, many of whom became close friends. I was fortunate to be part of a wonderful office. Arooj's generosity and treats brightened my days and reminded me of the importance of kindness. The check-ins with Safiya were often unexpected but always helpful; her resilience and dedication were really inspiring. I was lucky to reconnect with friends from my master's, Sasi and Dini, with whom I had lots of fun and created cherished memories. Sasi effortlessly helped me connect with others and have a good time. Her positivity was truly contagious: Talking to her was always uplifting and a source of much-needed support. Towards the end of my PhD, spending time with Alberto was both motivating and fun, and I learned so

much from the technical discussions I had with Geraldo.

My journey was also greatly enriched by my friendships outside AIPT. I was fortunate to connect with friends also pursuing PhDs at the university, with whom I became very close. The friendship I found in Monse and Lucas brought a sense of home. Carmen and Marta, from another institute, provided sincere support and were always a joy to be around. Being gym buddies with Daniel made a great difference during my PhD, and our coffee chats were a source of encouragement. The insights gained from these discussions were instrumental in developing the models presented in this thesis.

I was also lucky to join the team of resident advisors at Aston University. I am grateful for the opportunity to work with outstanding PhD students from whom I learned so much. Serving the student community on campus with the team has been a deeply meaningful experience.

Last but not least, I am indebted to my family: my parents, Nelson and Ana, and my sisters, Ana Eunice and Raquel, who, from thousands of miles away, cheered me on and were always there for me.

List of Publications

Publications Arising from this Thesis:

[12] N. Castro and S. Sygletos. A novel learned Volterra-based scheme for time-domain nonlinear equalization. In *Conference on Lasers and Electro-Optics*, page SF3M.1, San Jose, California, 2022. Optica Publishing Group. ISBN 978-1-957171-05-0.

[13] N. Castro and S. Sygletos. Learned Volterra equalization for WDM systems. In *2023 Asia Communications and Photonics Conference/2023 International Photonics and Optoelectronics Meetings (ACP/POEM)*, pages 1–4, 2023.

[16] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Design of time-domain learned Volterra equalisers for WDM systems. In *2024 International Conference on Optical Network Design and Modeling (ONDM)*, pages 1–3, 2024. doi:10.23919/ONDM61578.2024.10582691.

[14] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Design aspects of frequency-domain learned MIMO Volterra equalisers. In *CLEO 2024*, page JTu2A.87. Optica Publishing Group, 2024.

[15] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Field-enhanced filtering in MIMO learned Volterra nonlinear equalisation of multi-wavelength systems. In *ECOC 2024; 50th European Conference on Optical Communication*, pages 902–905, 2024.

[17] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Learned Volterra models for nonlinearity equalization in wavelength-division multiplexed systems. *Opt. Express*, 33(8):16717–16737, Apr 2025. doi:10.1364/OE.554077.

Additional Publications Completed During my Studies:

[29] A. Ellis, A. Ali, M. Tan, N. Salgado, and S. Sygletos. Mitigation of nonlinear effects in optical communications using digital and optical techniques. In *Optica Advanced Photonics Congress 2022*, page NeTu3D.4. Optica Publishing Group, 2022.

Contents

List of Figures

List of Acronyms

ADC analog-to-digital converter.

ANNs artificial neural networks.

ASE amplified stimulated emission.

ASIC application-specific integrated circuit.

AWGN additive white Gaussian noise.

BER bit-error rate.

CD chromatic dispersion.

CDE chromatic dispersion equalisation.

CNNs convolutional neural networks.

conv1D one-dimensional convolutional layer.

conv2D two-dimensional convolutional layer.

DBP digital backpropagation.

DFT discrete Fourier transform.

DNN deep neural network.

DSP digital signal processing.

EDFA erbium-doped fibre amplifiers.

FD frequency domain.

FE field-enhanced.

FFT fast Fourier transform.

FIR finite impulse response.

FWM four-wave mixing.

GN Gaussian noise.

GPU graphics processing unit.

IDFT inverse discrete Fourier transform.

IFFT inverse FFT.

IVSTF inverse Volterra series transfer function.

KD knowledge distillation.

L-IVSTF learned inverse Volterra series transfer function.

L-simIVSTF learned simplified inverse Volterra series transfer function.

LDBP learned digital backpropagation.

LSCO least squares constrained optimisation.

LSTM long short-term memory.

LTI linear time invariant.

MIMO multiple-input multiple-output.

ML machine learning.

MLP multi-layer perceptron.

MSE mean squared error.

NLE nonlinearity equalisation.

NLSE nonlinear Schrödinger equation.

NN neural network.

OFDM orthogonal frequency division multiplexing.

OPC optical phase conjugation.

PMD polarisation mode dispersion.

QAM quadrature amplitude modulation.

ReLU rectified linear unit.

RNNs recurrent neural networks.

RP regular perturbation.

RRC root-raised cosine.

SDM space division multiplexing.

SE spectral efficiency.

SGD stochastic gradient descent.

simIVSTF simplified inverse Volterra series transfer function.

SMF single-mode fibre.

SNR signal-to-noise ratio.

SPM self phase modulation.

SSF split-step Fourier.

SSMF standard single-mode fibre.

StpS steps per span.

TD time-domain.

UWB ultra-wideband.

VS volterra series.

VSNE Volterra series nonlinear equaliser.

VSTF Volterra series transfer function.

WDM wavelength division multiplexing.

XPM cross phase modulation.

List of Performance Metrics

The following metrics are used throughout this thesis to discuss and evaluate the performance of optical transmission systems:

- **BER:** The ratio of erroneous bits to the total number of transmitted bits:

$$\text{BER} = \frac{\text{Number of bit errors}}{\text{Number of received bits}}. \tag{1}$$

- **Effective SNR:** Obtained from the BER using the following relationship [119]:

$$\text{SNR} = 10 \cdot \log_{10}\left(\frac{2 \cdot (M-1)}{3}\left(\text{erfcinv}\left(\frac{\text{BER}\log_2(M)}{2} \cdot \left(1 - \frac{1}{\sqrt{M}}\right)\right)\right)^2\right), \tag{2}$$

  where M is the modulation order and erfcinv is the inverse of the complementary error function.

- **Q-factor:** Obtained from the BER as:

$$Q = 20\log_{10}\left[\sqrt{2}\text{erfc}^{-1}(2\text{BER})\right]. \tag{3}$$

Chapter 1

Introduction

Long-haul optical transmission links are essential to the Internet, carrying most of its data traffic. This traffic has been growing exponentially at a rate of around 60% [139], a trend expected to continue in the foreseeable future. Over the past three decades, technological advancements have enabled transmission networks to meet these increasing capacity demands [139]. The introduction of single-mode fibre (SMF) facilitated low-loss transmission in the C band. Optical amplifiers were developed to counter signal loss, allowing multiple wavelength channels (WDM) to be transmitted over a single fibre. The advent of coherent transceivers improved further the spectral efficiency through advanced modulation formats. Coherent receivers also enabled digital signal processing for the compensation of performance-limiting fibre impairments, such as chromatic dispersion (CD) and polarisation mode dispersion (PMD).

Transmission link capacity must continue to increase sufficiently to meet future demands. The Shannon-Hartley formula, which provides a lower bound for the capacity C, provides insights on which strategies can be used to increase it:

$$C \leq S \cdot B \cdot \log_2(1 + \mathrm{SNR}). \tag{1.1}$$

Here, B is the bandwidth, S is the number of spatial channels, and SNR is the signal-to-noise ratio. One option for increasing capacity is installing additional fibres, increasing the number of spatial channels S. Although this simple solution leads to capacity that increases linearly with the number of additional fibres, it is not cost-effective. Since each new fibre requires transceivers, amplifiers, and power, the system cost also rises linearly,

preventing any reduction in the cost per bit. This would also be the case in existing fibre networks, where activating dark fibres also requires new transceivers and power. Another approach is transmitting over wavelengths beyond the C and L bands, known as ultra-wideband (UWB) transmission. Despite significant progress, several issues remain before transmission beyond C+L becomes practical. Similarly to increasing fibre count, expanding to new bands would also require new amplification systems and transceivers, impacting cost. Additionally, the variations in fibre parameters with the transmission wavelength introduce increased nonlinear impairments, which has led researchers to consider alternative fibre types with lower loss and nonlinearity. Such development path would result in cost issues similar to installing more fibres for C-band transmission.

A promising solution lies in mitigating the impairments caused by fibre nonlinearity [5]. Fibre nonlinearity and amplified stimulated emission (ASE) limit the SNR, affecting the achievable capacity as per Eq. (1.1). Assuming fibre nonlinearity to be Gaussian and additive, the SNR in a transmission link can be expressed as [105]

$$\mathrm{SNR} = \frac{P}{P_{\mathrm{N}} + P_{\mathrm{S-S}} + P_{\mathrm{S-N}}}. \tag{1.2}$$

Here, P is the optical signal power, P_{N} is the total ASE noise, $P_{\mathrm{S-N}}$ is the total nonlinear signal-to-noise interaction and $P_{\mathrm{S-S}}$ is the signal-to-signal interaction.

Although modelling the nonlinear noise as Gaussian (and thus, non-deterministic) is a useful approximation for characterising system performance, the fibre nonlinearity giving rise to the $P_{\mathrm{S-S}}$ noise term in Eq. (1.2) is deterministic and can be partially suppressed. This suppression improves the SNR, producing a logarithmic increase in system capacity. Nonlinearity equalisation (NLE) is a key method for mitigating fibre nonlinearity [29]. It can be implemented using optical techniques such as optical phase conjugation (OPC) [74] or digital approaches [75, 85]. Digital NLE is particularly attractive as it avoids modifications to transmission infrastructure, which may reduce costs compared to other approaches. Despite extensive research, this approach has not found a clear path to commercialisation yet [107]. Arguably, this is primarily due to the unfavourable trade-off between computational complexity and performance that digital NLE currently entails.

A central approach for NLE is to emulate the inverse propagation of the received signal in the digital domain. DBP, the most extensively studied equalisation technique, achieves

this by employing the split-step Fourier (SSF) method. DBP has demonstrated versatility, with single-channel implementations addressing intra-channel impairments [72, 108, 96, 98] and full-band or MIMO implementations tackling inter-channel impairments [86, 93, 94]. Significant improvements have been achieved by adopting strategies to enhance accuracy, such as filtering in the nonlinear steps [108, 117], and the optimisation of the nonlinear step position [98]. These methods have reduced complexity, typically by reducing the required computational steps.

Recently, machine learning (ML) has gained attention for its potential in improving conventional NLE techniques. One notable development is learned digital backpropagation (LDBP), resulting from considering the optimisation of the DBP algorithm as a neural network [65]. While LDBP offers significant improvements in performance and cost reduction, its architecture is comprised of sequential linear and nonlinear operations which scale with the fibre link length, resulting in excessive processing latency and difficulty in leveraging hardware parallelism.

The Volterra series framework [104] presents a promising alternative to the SSF method on which DBP is based on, with inherent compatibility for parallel implementation. The framework can be used to develop equalisers that are a "parallel" counterpart of DBP. However, obtaining Volterra-based equalisers with comparable complexity to DBP requires truncating the series terms, heavily limiting precision and effectiveness. For example, third-order IVSTF equalisers, which gained attention for their fully-parallel, low-cost architecture, have not demonstrated comparable performance to DBP in single-channel equalisation [88], while their use for multichannel equalisation remains limited [137]. The slower progress in developing low-complexity Volterra-based equalisers compared to DBP is largely attributed to performance limitations resulting from the simplifications required for practical use.

This thesis revisits IVSTF-based equalisation with ML, addressing the accuracy limitations affecting its performance in single-channel and WDM transmission scenarios. The IVSTF is selected as the foundation of this work due to its inherent low-latency and low-complexity structure, which can potentially lead to efficient equalisers. By combining ML optimisation with the IVSTF, we aim to design explainable ML models [32] that outperform existing Volterra-based schemes. Our approach for achieving this is parameterising the IVSTF model and optimising it using gradient-based supervised learning [58]. Extensive numerical simulations are conducted to develop and evaluate the equalisers across various

transmission scenarios.

We first focus on single-channel equalisation, where the IVSTF is adapted into an ML model with improved accuracy in estimating SPM impairments. This effort results in a fully time-domain equaliser, the learned simplified inverse Volterra series transfer function (L-simIVSTF), which demonstrates a 3 dB SNR improvement over chromatic dispersion equalisation and matches the performance of LDBP [12]. This result highlights the adaptability of learned IVSTF-based models. The work is then extended to multichannel equalisation for WDM systems, leading to the development of learned MIMO equalisers that leverage the low-complexity and low-latency benefits of the IVSTF architecture. This work is facilitated by a novel computational framework for the design and training of the MIMO equalisers. We present three different equalisers: a fully parallel frequency-domain scheme (L-IVSTF), a field-enhanced (FE) version with improved adaptability (FE L-IVSTF), and a time-domain implementation (L-simIVSTF). A key contribution is the filtering strategy employed in the FE L-simIVSTF, integrating static and trainable linear stages to balance adaptability and computational efficiency. The equalisers introduced demonstrate unprecedented versatility for Volterra-based schemes, allowing multiple steps-per-span implementations and MIMO sizes of up to 9×9. In assessing equalisation capability, we first optimise each scheme to determine the best performance that can be achieved. The 9×9 L-simIVSTF and FE L-IVSTF equalisers demonstrate an average-per-channel SNR improvement of $\sim 2.2\,\mathrm{dB}$ over chromatic dispersion compensation, providing competitive performance compared to other theoretical studies on learned equalisers [124]. Our work then focuses on efficiency, examining the hyperparameter configurations that lead to the best performance-complexity tradeoff. Through a thorough comparison, the FE L-IVSTF emerges as the most efficient solution, providing a 1.7 dB average SNR improvement when implemented as a 9×9 model at 1 step per span.

1.1 Thesis Outline

The thesis is organised as follows:

- *Chapter 2* reviews key digital equalisation techniques for transmission systems. It covers two model-based methods: DBP and Volterra Series. The advantages and limitations of Volterra-based equalisers in terms of performance, complexity and flex-

ibility are discussed. This chapter also provides an overview of ML-based equalisers, differentiating between data-driven and model-driven approaches. Finally, the potential of model-driven approaches based on the Volterra series is explored.

- *Chapter 3* provides an overview of machine learning fundamentals, introducing basic concepts related to the design and training of artificial neural networks. The chapter also contrasts the design aspects of data-driven and model-driven approaches.

- *Chapter 4* outlines the methodology used to simulate optical transmission systems, including the digital signal processing techniques necessary for developing IVSTF-based single-channel and multichannel equalisers. The chapter also details the computational framework developed for implementing and training MIMO-learned equalisers.

- *Chapter 5* presents a novel learned Volterra scheme for single-channel equalisation. The performance of the scheme is evaluated in both single-channel and WDM transmission scenarios, comparing it with LDBP in terms of performance and complexity.

- *Chapter 6* presents novel learned MIMO equalisers for WDM systems, featuring three learned IVSTF-based models employing varied linear filtering techniques. The chapter details the optimisation of these equalisers and compares their performance and complexity to identify the model with the best tradeoff.

- *Chapter 7* gives concluding remarks on the work done in this thesis and offers future research directions.

Chapter 2

Literature Review

2.1 Nonlinear Impairments in Optical Transmission Systems

A key aspect of developing optical transmission systems has been overcoming fibre impairments [139]. Low-loss fibres transformed communications by enabling optical transmission over ultra-long distances, with the SSMF becoming the standard in modern optical networks. While fibre loss significantly degrades optical signals after a few tens of km, it is effectively countered by current amplification technologies, enabling transmission over thousands of kms. Some impairments introduced by SMFs become important in long-haul transmission. CD significantly impacts transmission as the high dispersion coefficient of the fibre results in a large accumulated dispersion. Fortunately, this CD can be corrected digitally at the receiver by applying static filters [113]. In contrast, the nonlinear impairments introduced are much more challenging to compensate for. The Kerr effect, which causes the fibre's refractive index to increase with the intensity of the optical field, gives rise to a nonlinear phase shift known as SPM. For a field $A(z)$ that has propagated over a length of fibre L, this shift is given by

$$\phi_{\mathrm{NL}} = \int_0^L P(z)dz = \gamma L_{\mathrm{eff}}|A(L)|^2. \tag{2.1}$$

Here, γ is the nonlinear parameter, L is the fibre length, L_{eff} is the effective fibre length [96], $P(z)$ is the signal's optical power, and z is the propagation distance. SPM primarily induces a frequency chirp on optical pulses that depends on the pulse shape. Moreover, this effect interacts with the dispersion of the fibre, leading to the broadening of the light

pulses.

In WDM systems, where multiple wavelength channels are transmitted over the SSMF fibre, the nonlinear phase shift experienced by each channel is influenced not only by the power modulation of its own field but also of the co-propagating channels. The shift induced by the other channels is known as **XPM**. Assuming the propagation of two channels with fields A_1 and A_2, the total nonlinear phase shift affecting field A_1 is

$$\phi_{\mathrm{NL}} = \gamma L_{\mathrm{eff}} \left(|A_1(z)|^2 + 2|A_2(z)|^2 \right). \tag{2.2}$$

The linear and nonlinear effects mentioned above do not affect light pulses independently, but interact with each other along the optical fibre as described by the NLSE:

$$\frac{\partial A(z,t)}{\partial z} = -\frac{\alpha}{2} A(z,t) + j\frac{\beta_2}{2}\frac{\partial^2 A(z,t)}{\partial T^2} - i\gamma |A(z,t)|^2 A(z,t). \tag{2.3}$$

Here, $A(z=0,t)$ is the field at the transmitter, β_2 is the group velocity dispersion parameter, and α is the propagation loss coefficient. The interactions between chromatic dispersion and nonlinearity described in this equation are intractable, making it difficult to compensate for nonlinear effects effectively.

The nonlinear impairments discussed above degrade the SNR in optical transmission systems by introducing amplitude and phase distortions to the field envelope. These distortions modify the signal's original modulation, causing the received waveforms to deviate from their expected values. Since the receiver interprets these deviations as noise, the noise floor described by the denominator of Eq. (1.2) is raised, limiting the usable signal power. Furthermore, as evidenced by Eqs. (2.1) and (2.2), the impact of SPM and XPM impairments is power-dependent. Figure 2.1 shows the performance, in terms of spectral efficiency, of a 20×100 km SSMF system with lumped amplification as a function of launch power. When using simple linear equalisation, the performance in the low-power regime increases linearly with the launch power, with the ASE noise as the main limiting factor. Conversely, in the high-power regime, performance declines with the launch power due to the increase in nonlinear fibre noise, which becomes the dominant limitation.

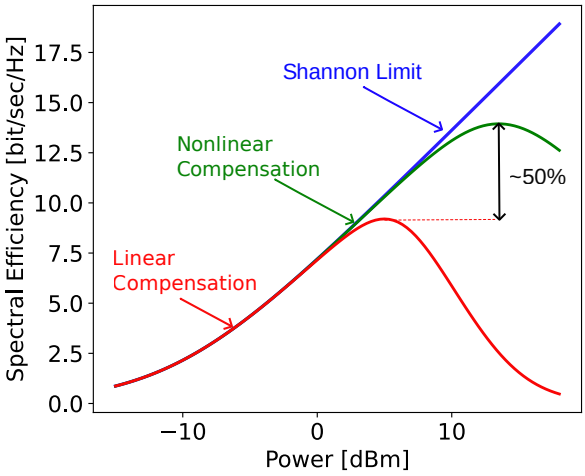


Figure 2.1: Spectral efficiency of a 20×100 km SSMF optical coherent system. Excessive launch power leads to suboptimal performance due to nonlinear noise. Optimal nonlinearity compensation yields approximately a 50% increase in capacity.

2.2 Digital Nonlinearity Equalisation for Optical Transmission Systems

Digital nonlinearity equalisation addresses fibre nonlinear impairments by applying algorithms to a digitally reconstructed signal, correcting distortions introduced during propagation [75]. Theoretical and experimental studies have extensively examined the potential gains of this technique in transmission systems [5, 30]. A common theoretical method for estimating the potential capacity improvements is employing the Gaussian noise (GN) model to consider the impact of nonlinear noise on capacity. The SNR in a transmission system after full-band compensation can be approximated as [5]

$$\mathrm{SNR} \approx \frac{P}{N_{\mathrm{span}} P_{\mathrm{ASE}} + 3\xi\eta P^2 P_{\mathrm{ASE}}}. \tag{2.4}$$

Here, N_{span} denotes the number of fibre spans, $\xi = N_{\mathrm{span}}(N_{\mathrm{span}} - 1)/2$, η is a nonlinear distortion coefficient, and P_{ASE} is the ASE noise introduced by each amplifier. Perfect full-band compensation removes the signal-to-signal noise term in Eq. (1.2). This SNR expression can then be employed to estimate the system capacity. Figure 2.1 illustrates how spectral efficiency improves with nonlinearity equalisation, resulting in approximately a 50% capacity increase. These enhancements can potentially improve operational margins for telecom operators, with theoretical models suggesting that compensating fibre nonlinearity

could reduce the total fibre count required for large-scale networks [29].

However, achieving the gains predicted by theory is difficult in practice. In practical transmission systems, where fibre capacity is exploited through WDM, the full compensation of nonlinear impairments requires access to the full bandwidth of the multiplexed signal. However, the bandwidth of digital receivers is limited by their analog-to-digital converters (ADCs), which restricts equalisers to a small portion of the total signal spectrum, diminishing the achievable gains. An additional practical constraint is the requirement for low computational complexity. With increasing concerns about power consumption in communications infrastructure, telecom providers favour low-complexity signal processing for their commercial solutions [133]. However, the approximations of the fibre link's nonlinear response employed by conventional compensation techniques are only accurate with a sufficient number of steps (in the case of DBP) or considered terms (in the case of perturbation-based methods), which often translates to high computational complexity. Restricting the computational cost of equalisers often prevents them from delivering satisfactory performance [86].

Given these limitations, single-channel equalisation, an approach for single-channel receivers, has been the primary focus of research. While it is the most feasible option for implementation in practical transceivers, its performance in multichannel transmission scenarios is limited by its inability to mitigate the dominant interchannel effects (yielding less than 1.5 dB Q^2-factor improvement [107]). MIMO schemes [93] have attracted attention, as they do not require wideband receivers but rather integrated single-channel receivers capable of sharing information. This equalisation approach aligns with research indicating that wavelength parallelism is necessary for increasing the capacity of optical networks [139]. The limited capacity of wavelength channels is driving efforts to integrate parallel coherent transceivers onto a single chip, which may enable the future practical implementation of MIMO nonlinearity equalisation.

In the following sections, we examine the main nonlinearity equalisation approaches for transmission systems, highlighting the gaps in the existing literature that this thesis addresses. We categorise these approaches into **conventional** and **ML-based** techniques. In particular, the limitations of conventional methods are discussed, along with how ML has been leveraged to overcome some of these challenges. Finally, we discuss how this thesis builds on these advancements.

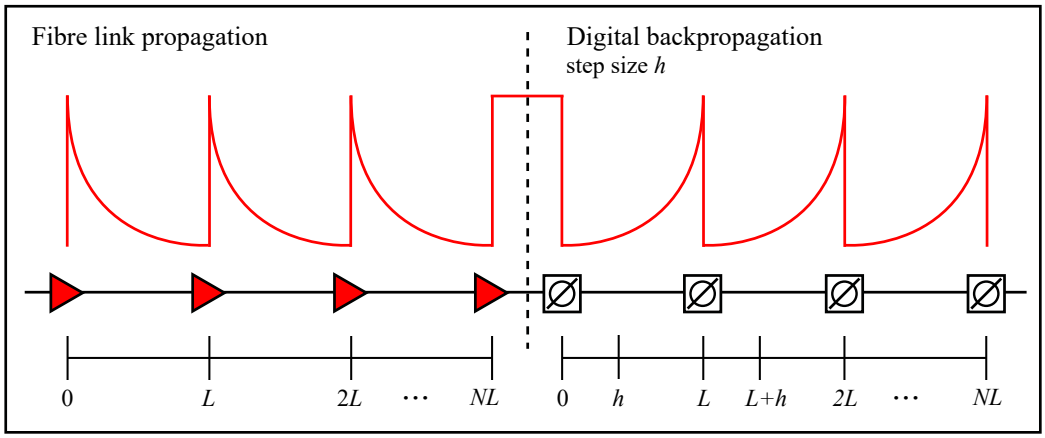


Figure 2.2: Visual representation of a physical forward propagation link and a virtual backpropagation link implementing DBP with a step size h. The red curves represent the corresponding signal power evolutions.

2.2.1 Conventional Techniques

We call conventional equalisation techniques those based on solving the NLSE using numerical methods, obtaining an inverse model to cancel nonlinear impairments. This section examines two conventional techniques: DBP and the Volterra Series. We provide an overview of the algorithms within each technique and review their use for single-channel and multichannel equalisation for transmission systems.

Digital Backpropagation

DBP simulates the reverse propagation of the signal through the optical fibre link. This is achieved by defining a virtual link with inverse parameters that sufficiently approximates the propagation effects of the forward link [72], as illustrated in Fig. 2.2. DBP employs the SSF method to solve the inverse propagation equation, dividing it into linear and nonlinear components and solving them separately. This approach assumes that the interaction of linear and nonlinear effects over sufficiently short propagation distances is negligible. Consequently, an effective algorithm implementation requires subdividing the transmission link into short segments, which are addressed sequentially.

The NLSE (Eq. (2.3)) can be rewritten as

$$\frac{\partial A}{\partial z} = (\hat{D} + \hat{N})A, \tag{2.5}$$

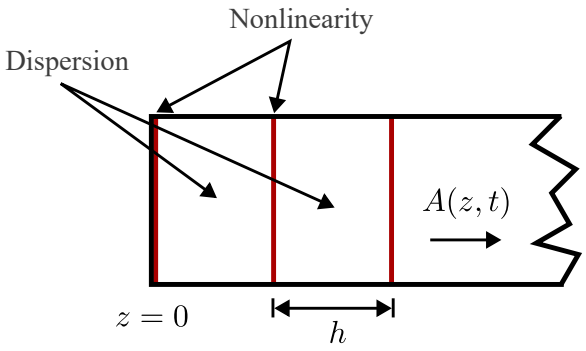


Figure 2.3: Illustration of the split-step Fourier method solution to the NLSE. $A(z,t)$ is the propagated field. After each nonlinear phase shift, a dispersive step of length h is applied.

where the $\hat{D}$ and $\hat{N}$ operators are given by

$$\hat{D} = -\frac{i\beta_2}{2}\frac{\partial^2}{\partial T^2} \tag{2.6}$$

and

$$\hat{N} = \frac{\alpha}{2} + i\gamma|A|^2. \tag{2.7}$$

Over a length of fibre h, the SSF solution can be expressed as two steps, as shown in Fig. 2.3. In the first step, only nonlinearity is considered and $\hat{D} = 0$. In the second step, only dispersion is accounted for, and $\hat{N} = 0$. The field at $z = h$ can be expressed as

$$A(z+h,T) \approx A(z,T)e^{h\hat{D}}e^{h\hat{N}}. \tag{2.8}$$

Reducing the step size h improves the approximation of the solution and translates into better nonlinearity compensation. However, this comes at the cost of increased computational effort which is an obstacle to the practical implementation of the algorithm. The SSF method resolves the linear components in the frequency domain (FD) and the nonlinear ones in TD. This dual-domain approach has been widely adopted for the implementation of DBP equalisers, since performing linear steps with fast Fourier transform (FFT) has been considered the most computationally efficient approach. In this case, the computational cost of DBP mainly stems from the FFT. However, linear steps can also be performed in the time domain by approximating the dispersion transfer function using finite impulse response (FIR) filters, which has the advantage of avoiding repeated Fourier transformations

and facilitates real-time processing [85]. For instance, time-domain DBP implementations suitable for application-specific integrated circuit (ASIC) have been proposed [39]. A step of TD DBP can be formulated as

$$A(z+h,t) = (A(z,t) * h_{\mathrm{CD}}) \cdot e^{\alpha h/2} \cdot e^{-jh\gamma|A|^2}, \tag{2.9}$$

where h_{CD} is an FIR filter addressing the dispersion of step h. Time-domain DBP algorithms could potentially offer competitive computational cost [90] as convolution operations can be more efficient than frequency-domain filtering if the filters are sufficiently short [38]. However, this depends on the transmission scenario, as the filter length is constrained by the memory of the transmission channel, which sets a lower bound on the filter length. Additionally, FIR filters introduce inaccuracies due to the truncation of the dispersion impulse response, which quickly worsens as the filter length is shortened. Therefore, significant efforts have been directed towards developing methods that use short filters while maintaining accuracy. One such method is the joint optimisation of FIR filter pairs, which has been shown to significantly reduce filter length [145]. Notably, time domain implementations have been useful for the parameterisation of DBP algorithms for ML optimisation, as discussed in section 2.2.2.

The DBP algorithm has been adapted to various transmission scenarios, from single-channel setups [72] to multi-wavelength systems [85, 89]. This is done by solving an adequate propagation equation. In single-channel transmission, the propagation equation accounts for fibre losses, CD and SPM. For dual-polarisation signals, solving coupled equations to consider polarisation interactions yields a suitable equaliser[70]. In multichannel systems, a **full-band DBP** equaliser [86] can be derived by solving a single NLSE that describes the propagation of the entire multiplexed signal. Alternatively, by solving coupled NLSE each channel can be backpropagated separately, leading to a **MIMO DBP** equaliser where XPM interactions between co-propagated channels are explicitly accounted for [26]. These approaches involve varied computational and implementation requirements.

Single-channel DBP has been extensively investigated and is currently the primary algorithm for benchmarking nonlinearity equalisation performance. Its flexibility enables its application in dispersion-unmanaged and dispersion-managed links [98]. However, its computational cost has prevented it from becoming a commercial solution. Therefore,

significant efforts have focused on reducing the hardware complexity of DBP. The main approach to pursue this has been to reduce the number of required DBP steps. This has been achieved by modifying the nonlinear steps to consider the impact of neighbouring symbols [108][117]. Although this increases the cost of nonlinear steps, it leads to overall complexity reduction. Another method, proposed for the processing of single-channel wideband signals, involves backpropagating slices of the received signal spectrum via subband processing [73, 19].

Full-band DBP processes WDM channels as a unified field. While it effectively addresses intra and interchannel impairments with essentially the same algorithm as single-channel DBP, it is much more computationally demanding. In this approach, the separate treatment of linear and nonlinear effects done by the SSF method requires a much shorter dispersive length to be effective. This requirement is imposed by the interchannel impairments. If the number of steps is too low, the algorithm performs worse than simple chromatic dispersion equalisation. Additionally, digital oversampling rates higher than those needed for single channel equalisation are required to process a wider bandwidth [86]. These requirements increase with the bandwidth of the signal.

Unlike full-field DBP, MIMO DBP is applied after demultiplexing the channels and processes them jointly. The implementations proposed initially require a high number of steps to adequately represent the interaction of the walkoff delay between channels with fibre nonlinearity [91]. As with full-field DBP, this leads to much higher step counts than single-channel DBP. To reduce the number of steps, frequency-domain filtering in the nonlinear stages can be used to improve the estimation of dispersed nonlinearity [93, 18].

The optimisation of single-channel and multichannel DBP with ML techniques is covered in section 2.2.2.

Volterra Series

The Volterra series is a mathematical framework that effectively models nonlinear interactions in systems with "memory" effects [115], where past inputs influence the current output. The method decomposes the nonlinear response of a system into a series of terms. It offers a more detailed alternative to the Taylor series for analysing complex systems and has demonstrated its applicability across many engineering fields. One of its main applications is the modelling and equalisation of nonlinear distortion. Volterra equaliser implementations can

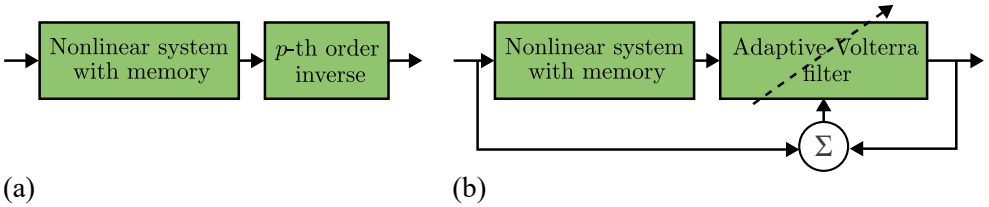


Figure 2.4: Compensators based on the Volterra inverse for a nonlinear system: (a) analytical compensator, (b) adaptive compensator.

be categorised as **analytical** or **adaptive** [134]. These two approaches are illustrated in Fig. 2.4.

When implemented as an adaptive filter, parameters are not analytically calculated but instead dynamically adjusted to minimise a defined error criterion, such as the mean squared error (MSE). Adaptive Volterra filters are flexible and can be implemented in the time or frequency domain. The output y of a P-th order discrete-time Volterra filter with memory M can be expressed as a function of an input x as [130]:

$$y(k) = w_{dc} + \sum_{r=1}^{P} \sum_{k_1=0}^{M-1} \cdots \sum_{k_r=k_r-1}^{M-1} w_r(k_1, k_2, \ldots, k_r) \times x(k-k_1) \cdots x(k-k_r). \tag{2.10}$$

Here, w_r are the r-th order Volterra kernels. While adaptive Volterra filters are a promising solution for compensating transceiver nonlinearity in short-reach systems [6, 130], their use in equalising Kerr nonlinearity in long-haul systems is limited [102]. The number of coefficients in the Volterra filter grows as $\mathcal{O}(M^P)$, which leads to an extremely high complexity if P or M are large. Even for low-order equalisers, the large channel memory in transmission scenarios requires a filter with a large M, resulting in a prohibitively high complexity [23].

In contrast, analytical Volterra equaliser implementations are promising for transmission systems. A central analytical tool for developing analytical equalisers, based on the frequency-domain formulation of the Volterra series, is the **Volterra series transfer function (VSTF)**, which describes the relationship between the output and input of a system as [115]

$$\begin{aligned} Y(\omega) = \sum_{n=1}^{\infty} \int \cdots \int H_n(\omega_1, \ldots, \omega_{n-1}, \omega - \omega_1 - \cdots - \omega_{n-1}) X(\omega_1) \cdots X(\omega_{n-1}) \\ \times X(\omega - \omega_1 - \cdots - \omega_{n-1}) d\omega_1 \cdots d\omega_{n-1}, \end{aligned} \tag{2.11}$$

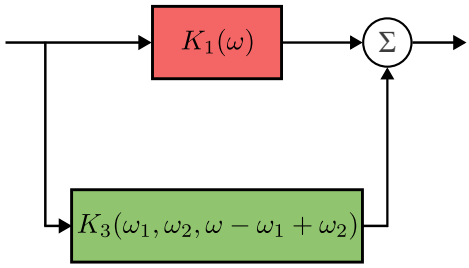


Figure 2.5: Single-channel, single-polarisation compensator based on the third-order IVSTF. $K_1(\omega)$ and $K_3(\omega)$ are the first and third order inverse kernels, respectively. Nonlinear distortion is modelled as an additive term.

where $H_n(\omega_1, \omega_2, \cdots, \omega_n)$ is the n-th order frequency domain Volterra kernel, and $Y(\omega)$ and $X(\omega)$ are the Fourier transforms of the output and input, respectively. The VSTF is employed for modelling nonlinear signal propagation in optical fibres, providing an alternative to the split-step Fourier method. A VSTF for single-mode optical fibres is presented in [104], where analytical expressions for each kernel up to the 5-th order are found. This method parallels the regular perturbation (RP) approach [136], in which the solution to the NLSE is formulated as a power series of the nonlinear coefficient γ. The RP method provides closed-form approximations of the output field equivalent to those obtained using the VSTF [136].

The VSTF fibre model attracts attention for its potential in equalising distortions in optical fibre links. However, its application faces challenges due to the computational cost of calculating the complex integrals in Eq. (2.11) [104]. To address this, researchers often rely on truncations to the third-order term of the Volterra series, expressing the solution to the NLSE as:

$$\begin{aligned} A(\omega, z) \approx\ & H_1(\omega, z)A(\omega) \\ & + \int\int H_3(\omega_1, \omega_2, \omega - \omega_1 + \omega_2, z)A(\omega_1)A^*(\omega_2) \times A(\omega - \omega_1 + \omega_2)d\omega_1 d\omega_2, \end{aligned} \tag{2.12}$$

where $A(\omega)$ is the Fourier transform of the electrical field at the fibre input ($z = 0$). A compensator for the distortions described by the VSTF model can be derived by obtaining its p-th order inverse, which nullifies the VSTF kernels up to the p-th order when put in cascade with it [115]. This process produces inverse kernels $K_1(\omega)$ and $K_3(\omega_1, \omega_2, \omega - \omega_1 +$

ω_2), which for an N-span link are

$$K_1(\omega) = e^{j\beta_2 N L_{\text{sp}} \omega^2/2} \tag{2.13}$$

and

$$K_3(\omega) = \frac{j\gamma}{4\pi^2} K_1(\omega) \times \frac{1 - e^{-(\alpha + j\beta_2 \Delta\Omega) L_{\text{sp}}}}{\alpha + j\beta_2 \Delta\Omega} \sum_{k=1}^{N} e^{jk\beta_2 L_{\text{sp}} \Delta\Omega}. \tag{2.14}$$

Here, L_{sp} is the length of a fibre span, N is the total number of spans, and $\Delta\Omega = (\omega_1 - \omega)(\omega_1 - \omega_2)$. These inverse kernels are applied to the Fourier transform of a received field $A(z,t)$ using Eq. (2.12). A schematic of a third-order compensator implemented with these kernels is shown in Fig. 2.5. Ignoring the distortion inside of fibre spans leads to a simplified third-order kernel in which loss is decoupled from dispersion:

$$K_3(\omega) \approx \frac{j\gamma}{4\pi^2} \times \frac{1 - e^{-\alpha L_{\text{sp}}}}{\alpha} K_1(\omega) \sum_{k=1}^{N} e^{jk\beta_2 L_{\text{sp}} \Delta\Omega}. \tag{2.15}$$

The summation in (2.15) shows that the K_3 kernel can be divided into separate stages for each span, which may be computed in parallel. Equalisers based on these kernels often use discrete frequency domain processing, defining the kernels in the discrete frequency domain and applying them to the signal's discrete Fourier transform (DFT). The necessary DFTs are computed efficiently using FFT algorithms. Further simplifications may be achieved by ignoring inter-frequency-mixing between the frequency-shifted fields in the "triplet" from Eq. (2.12), allowing its separate time-domain computation. The inverse kernels are then realised in a structure known as the **IVSTF**, implemented with sequential time-frequency domain steps, as shown in Fig. 2.6 [88]. In the figure, H_{CD} is the CD transfer function corresponding to a single fibre span applied in the frequency domain. The operation $jc|\cdot|^2(\cdot)$ is applied in the time domain, where c represents a coefficient including both fibre loss and nonlinearity. This approach enables a fully parallel architecture and reduces complexity compared to the original VSTF fibre model but delivers lower performance than DBP when employing the same number of computational steps per span, achieving 1 dB improvement over chromatic dispersion equalisation (CDE) compared to DBP's 1.7 dB [88]. Furthermore, the accuracy of the IVSTF is only acceptable in the "quasi-linear" power regime, showing unsatisfactory performance in the "highly nonlinear" regime. This accuracy limitation is

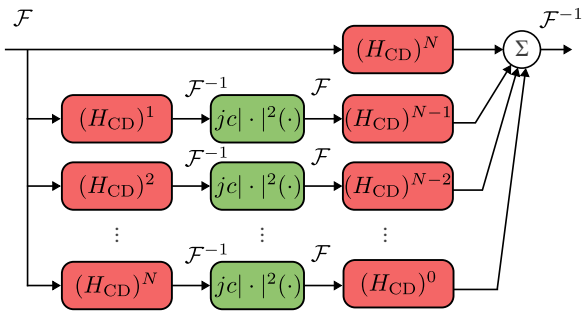


Figure 2.6: Single-polarisation IVSTF scheme for an N-span fibre link, as proposed in [88]. H_{CD} is the CD transfer function corresponding to a single span of fibre, c is a constant proportional to the fibre nonlinear parameter and the effective length of the fibre span.

explained by how the nonlinear phase shift is modelled: it is a truncation of a polynomial expansion of the nonlinear phase shift obtained from the SSF method. Schemes based on truncations to the fifth order could improve performance. However, they are much more complex than third-order models, and their demonstrated improvement is limited to single-channel transmission scenarios [2].

Another third-order scheme, known as the **Volterra series nonlinear equaliser (VSNE)** [55] employs the discrete Fourier transform to apply the nonlinear kernel to the signal as follows:

$$A_{\mathrm{EQ}}(\omega_n) \approx \sum_{i=1}^{N_{\mathrm{FFT}}} \sum_{j=1}^{N_{\mathrm{FFT}}} K_3(\omega_i, \omega_j, \omega_n - \omega_i + \omega_j) A(\omega_i) A^*(\omega_j) \times A(\omega_n - \omega_i + \omega_j), \tag{2.16}$$

where $A_{\mathrm{EQ}}(\omega)$ is the DFT of the equalised field, N_{FFT} is the FFT size, and i and j are auxiliary indices. Compared to the IVSTF, this method performs the integration in (2.12) over the entire nonlinear kernel, capturing dispersion-nonlinearity interactions more accurately. Consequently, the VSNE scheme has demonstrated superior performance to comparable DBP implementations while requiring less computational effort. Further developments have investigated complexity reduction strategies by simplifying the frequency-domain kernels [54] and using fully time-domain implementations [56]. However, practical adoption is limited due to higher implementation complexity than the IVSTF. For example, the summation over the frequency components of the third order kernel in Eq. (2.16) requires more intricate block processing than the simple CD filtering and element-wise nonlinear calculations depicted in Fig. 2.6.

Initial investigations into Volterra equalisers focused on single-channel transmission. While the extension of third-order Volterra-based models for dual-polarisation systems was straightforward [88], extending these models for multichannel transmission has seen limited progress. Although the Volterra series has been successfully applied to analyse XPM and four-wave mixing (FWM) inter-channel impairments [110], their use for the equalisation of these impairments has remained under-explored. While various multichannel DBP implementations have been developed [92, 93, 94], this versatility has not yet been demonstrated for the Volterra approach. The main advancement for multichannel transmission systems has been a full-field scheme for super-channel systems based on the third-order IVSTF [137] (employing the same structure from Fig. 2.6), which has shown lower complexity than its DBP counterpart while delivering a similar performance. Yet, no MIMO Volterra alternatives have been developed. Nevertheless, IVSTF-based MIMO schemes have been investigated in coherent space division multiplexing (SDM) systems [138], indicating the potential for similar methods in WDM scenarios.

2.2.2 ML-based Techniques

The past decade has seen remarkable advances in ML, with the performance of ML models in areas such as computer vision [62] and natural language processing, prompting interest in applications to optical communications. In this field, the equalisation of signal impairments is one of the main tasks where the potential of ML is being investigated [97]. The task of removing deterministic distortion from a signal is compatible with the capabilities of a range of ML models. For instance, the temporal dependencies in transmission data make the problem well-suited to mature ML approaches such as time-series forecasting methods [82]. Supervised learning strategies are particularly useful in this context since training data (transmitted and received symbols) may be available to provide an error signal for ML optimisation, in line with the established data-aided approach to DSP [80]. Furthermore, ML-based equalisation aligns with current research efforts to make networks more dynamic and adaptive [1]. ML models generally do not require the knowledge of transmission parameters, unlike conventional equalisation schemes, potentially simplifying their configuration. Finally, they could potentially provide better performance-complexity trade-offs than conventional approaches. Compression techniques such as pruning [10], quantisation [50] and clustering [59] may enable low-complexity implementations [46], making deployment feasible

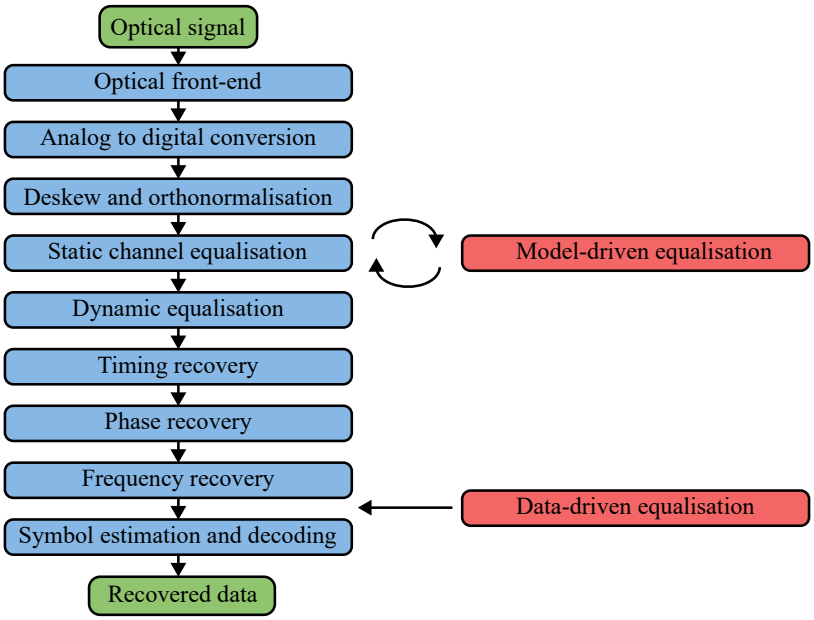


Figure 2.7: Block diagram of a coherent receiver architecture, illustrating the integration of model-driven and data-driven equalisation techniques within the signal processing chain. Model-driven equalisation replaces the static equalisation stage, whereas data-driven equalisation is usually placed at the end of the DSP chain.

in resource-constrained environments such as optical transceivers [40].

However, the potential adoption of ML-based nonlinear equalisers depends on successfully integrating with the existing DSP. In digital coherent receivers, nonlinearity equalisation is typically implemented as a discrete stage within the DSP chain [114]. The entire DSP architecture is traditionally grounded in the physics of optical fibre links [60] and comprises various subsystems, as depicted in Fig. 2.7. Deterministic fibre impairments are addressed within the static channel equalisation subsystem through linear and nonlinear compensation techniques. A typical method for incorporating ML-based equalisation involves omitting the conventional equalisation block and appending an artificial neural network NN to the chain [68]. However, this approach diverges from the established physics-based design paradigm that underpins the DSP stack. Additionally, ML-based equalisers must demonstrate the ability to meet the latency and reliability demands of optical transmission systems. These considerations have steered recent developments towards interpretable ML-based DSP solutions, which, instead of discarding existing DSP stages, implement them as neural network layers to optimise their parameters [64, 100].

Two main approaches can be distinguished in the academic literature on ML-based nonlinearity equalisation: **data-driven** and **model-driven**. Data-driven schemes use black

box architectures and rely on large amounts of training data, without explicitly modelling the physical properties of the fibre. Examples are standalone equalisation stages based on NNs [123, 42]. Model-driven schemes, on the other hand, incorporate the physical principles governing signal propagation in their architecture. Examples of this approach are inverse models for equalisation based on DBP and perturbation theory, which are adapted to be trained using gradient-based techniques [65, 100]. While this distinction is not yet a standard convention in the field, it provides a useful framework for the work in this thesis. It is important to note that these categories are not rigid since it is possible to infuse domain knowledge into NN design to varying degrees. For instance, the method in [144] applies feature engineering employing perturbation theory to feed triplets to a conventional neural network, blurring the line between data-driven and model-driven designs.

Each approach has distinct requirements for algorithm integration and learning paradigms, leading to varying training procedures and integration into DSP architectures. For instance, the stage's placement in the DSP pipeline varies between these approaches, as they generally have different sampling rate requirements. This is illustrated in Fig. 2.7. Data-driven schemes operate at baud rate and thus are typically located at the end of the DSP pipeline [99]. In contrast, model-driven approaches usually require higher processing rates and are therefore placed earlier in the chain. The preprocessing overhead and data formatting requirements also vary between these approaches. Data-driven models typically require pre-processing steps such as windowing to prepare data for model input [123]. In contrast, model-driven approaches are transparent to the data flow of conventional DSP, maintaining the input dimensionality in the output batches [35]. Another distinction lies in the learning paradigm underpinning each approach. Data-driven approaches can be implemented as regression or classification learning tasks [44], while model-driven schemes are limited to regression-based learning. The following sections discuss these approaches, outlining their advantages and limitations. We begin with a review of the research on data-driven schemes, followed by an examination of the model-driven approach, which was used to develop the models presented in this thesis.

Data-driven Equalisation Schemes

Various data-driven techniques have been investigated for mitigating nonlinear impairments in optical transmission systems, including decision trees [118], support vector machines, and

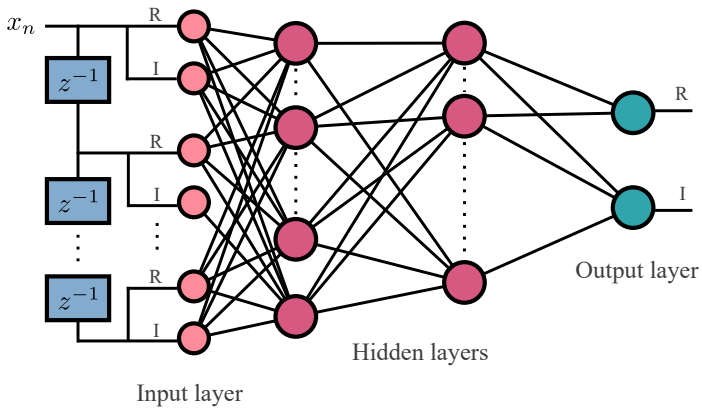


Figure 2.8: Neural network equaliser proposed in [123]. Delay blocks are used at the input to account for the channel memory effect. The number of neurons of the hidden layers are hyperparameters to optimise. The output layer has one neuron for each component of the symbol to be predicted.

NNs [97]. Among these, NNs have received the most attention due to their extraordinary efficacy in domains like computer vision [83]. Their applicability is supported by the Universal Approximation Theorem [63], which states that NNs can approximate any function, suggesting their suitability for addressing fibre-based impairments.

Initial investigations of NN-based nonlinearity equalisation in coherent optical systems employed NNs in single-channel scenarios [123]. This approach attracted interest due to its straightforward architecture, shown in Fig. 2.8. While NNs are not inherently suited to processing temporal data sequences, learning inter-symbol dynamics is achieved by feeding sample sequences to the model using sliding-window preprocessing. The depicted scheme is used for regression, with the last layer having a separate activation for the real and imaginary parts of the symbol. These models have shown competitive performance under constrained complexity [42]. However, a significant limitation is their susceptibility to over-fitting [45], requiring careful design [31]. The application of more advanced neural networks has also been proposed, leveraging their respective strengths in data processing. Convolutional neural networks (CNNs), recognised for their pattern extraction capabilities, and recurrent neural networks (RNNs), designed to capture temporal dependencies, have been explored. Long short-term memory (LSTM) networks, a specialised type of RNN with improved ability to handle large data sequences, have proven particularly effective for managing the large channel memory typical of coherent transmission systems[22]. The architecture of a single channel, dual-polarisation RNN equaliser is shown in Fig. 2.9 (a). The input features are the polarization components of the channel. This scheme is also

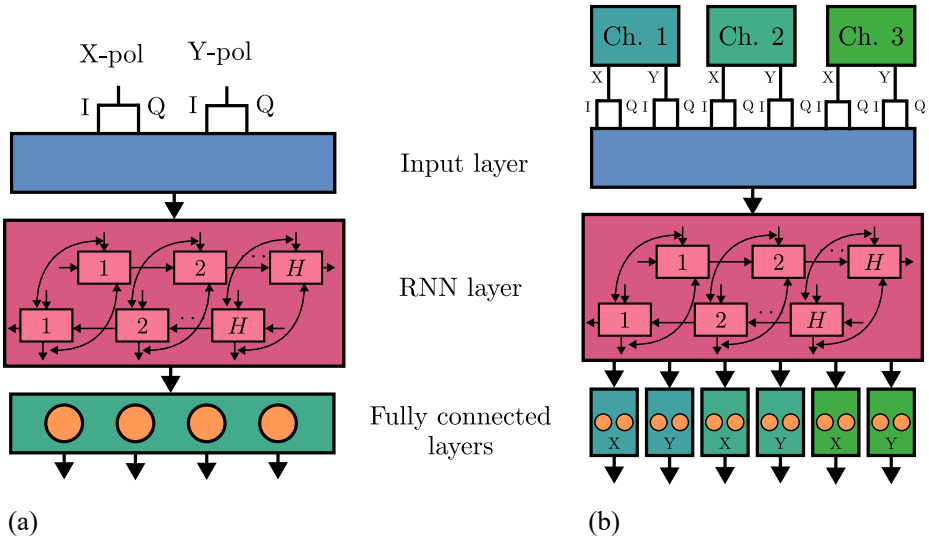


Figure 2.9: (a) Single channel [23] and (b) multichannel [24] RNN equalisers. Both models are comprised by an input layer, a bi-directional RNN layer, and a fully connected layer. The number of units of the fully connected layers corresponds to the real and imaginary parts of the symbols predicted for each channel.

regression-based, with output units for each symbol component of each polarization.

Comparative studies have assessed these models, including hybrid approaches combining convolutional, recurrent and dense layers [43, 120]. Among these models, recurrent-layer architectures have shown superior performance. In numerical simulations, LSTM-based equalisers have achieved a 1.3 Q-factor improvement in a 9×110 km system, surpassing the performance of 3 StpS DBP [46]. In contrast, experimental validation on a link of the same length demonstrated a more modest increase of 0.7 dB. Despite these clear advancements, practical use of RNN models faces significant challenges. The computational complexity required for inference is substantial, posing a barrier to their practical implementation. Additionally, the recurrent connections between cells cannot be parallelised, hindering their integration into hardware. The complexity issue has been addressed through compression techniques such as dropout regularisation and network pruning, reducing inference costs [112] [47] [49] and even making RNN equalisers less computationally demanding than single-channel DBP implemented at 1 StpS. However, parallelisation limitations have prompted researchers to return to feed-forward architectures for practical implementations. In this direction, Knowledge distillation (KD) has been recently proposed as a means of transferring knowledge from RNNs to simpler non-recurrent models [129].

Despite promising results in single-channel scenarios, performance of single-channel NN

equalisers in WDM transmission systems remains limited due to their inability to learn inter-channel impairments, prompting the investigation of multichannel equalisation approaches. Early work employed NNs in MIMO implementations, processing multiple orthogonal frequency division multiplexing (OFDM) channels to address FWM [51]. Given the strong performance of RNNs in single channel environments, they have also been investigated for MIMO equalisation in WDM systems. Figure 2.9 (b) shows the architecture of a 3-channel RNN equaliser proposed in [24], which retains the general architecture from the single-channel equaliser. The difference is in the input features, which now accommodate the polarization components of each channel. This equaliser, without the need for compression techniques, exhibit lower per-channel complexity compared to its single-channel RNN counterpart. However, its effectiveness is limited as they do not scale well beyond a limited number of processed channels [24]. Furthermore, it has the same parallelisation limitations as single-channel RNN equalisers. Consequently, recent efforts have investigated addressing inter-channel impairments without using MIMO equalisation. A single-channel equaliser based on multi-task learning has shown potential in compensating for XPM impairments [128]. However, the reported improvements remain limited.

Finally, the adoption of data-driven approaches raises practical concerns. In both design and deployment, neural networks are often treated as black boxes, requiring minimal understanding of the internal mechanisms behind their predictions. The inability to explain model outputs is problematic for network operators [11], concerned with network management and troubleshooting. Additionally, the relationship between the topology of data-driven models and their performance on a given task is poorly understood, forcing practitioners to rely on experience and heuristics when selecting architectures. As an example, the design of the RNN equaliser in [24] employed a grid search to choose the optimal number of hidden units, without the possibility of validating these decisions against physical explanations. Yet another challenge is the prevalent risk of overestimating the benefits of NN-based equalisation, since ensuring that the model has learned only the intended impairments is difficult. Although studies provide detailed recommendations to avoid training pitfalls [44], adherence to these guidelines cannot guarantee that the models will not overfit to the data or behave unpredictably, particularly under the varying operating conditions that could be encountered in deployment. This unpredictability might be unacceptable in telecommunication systems, which demand an extremely low error tolerance.

Model-driven Equalisation Schemes

In contrast with data-driven schemes, the **model-driven** approach [143] integrates physics models with machine learning optimisation techniques to produce more efficient and interpretable schemes. This is usually done by designing the topology of a model considering underlying physics laws, such as approximate solutions to propagation equations. This approach to equaliser design offers several advantages. It provides clear hyper-parameter choices, potentially including the type and number of layers, neurons and activation functions. Additionally, it may reduce the need for training data, as some of the knowledge the model would otherwise need to learn is embedded in the model itself. These advantages have made the model-driven design approach attractive for applications in the physical layer of communication systems [61]. Decades of research in this domain have yielded extensive domain knowledge, offering well-established models as a basis for the schemes. In the context of equalisation in optical transmission systems, the available models of well-characterised fibre impairments, as well as DSP techniques for coherent transceivers, are assets for model-driven design. Another factor favouring this approach is the limited availability of training datasets in this area. Data privacy and proprietary information concerns restrict access to real-world transmission data, leading to the exploration of alternatives to conventional data-hungry ML approaches. Requiring less training data could make model-driven methods a more attractive solution for vendors.

A potential drawback, however, is that model-driven schemes are often initialised with parameters corresponding to the physics-informed topology, which in turn depends on knowledge of the transmission link. This initialisation approach is intended to ensure the optimal convergence of the algorithm. In contrast, black-box models are fully agnostic to system parameters, which can make them more suitable for dynamic environments. Nevertheless, alternative initialisation methods that do not require knowledge of the link have been proposed, such as sampling values from a probability distribution, although these appear prone to sub-optimal convergence.

Learned Digital Backpropagation

LDBP has become the central model-driven ML approach for nonlinearity equalisation in transmission systems. This technique, introduced in [66], was inspired by the observation

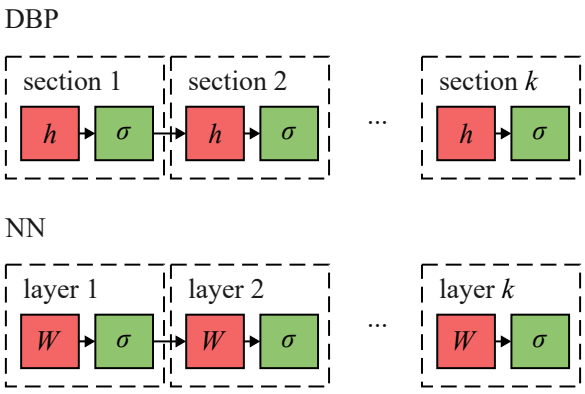


Figure 2.10: Comparison between the functional forms of DBP and NN as noted in [9]: h represents the linear steps in DBP (implemented in the time domain with FIR filters) and W the weights in NNs. Both models feature pointwise nonlinear activations σ.

that the DBP algorithm shares a similar functional form with neural networks, as illustrated in Fig. 2.10. LDBP leverages ML optimisation to improve the efficiency of DBP equalisation. In a process known as "deep unfolding" [4], the iterations of DBP are unfolded into a layer-wise structure, with key parameters from the algorithm stages set as trainable, which are then optimised using ML techniques. While conventional DBP implementations apply identical, non-adaptable filters across the backpropagation steps [72], resulting in the accumulation of the numerical errors and the introduction of wave-mixing distortions [35], LDBP mitigates these issues by jointly optimising the combined response of the filters of the algorithm. An LDBP model with M steps (or layers) can be formulated as [58]

$$\boldsymbol{f}(x) = \boldsymbol{\sigma}(\boldsymbol{A}^{(M)} \dots \boldsymbol{\sigma}(\boldsymbol{A}^{(1)}\boldsymbol{x})). \tag{2.17}$$

Here, $\boldsymbol{A}$ are matrices associated with the linear steps of the SSF method, and $\boldsymbol{\sigma} = xe^{j\gamma z|x|^2}$ is an element-wise operator. Different versions of LDBP can be obtained depending on which variant of the SSF is parameterised (for example, asymmetric or symmetric), how DBP's linear and nonlinear stages are implemented —either in time or frequency domain —and which parameters are set as trainable. The introduction of trainable parameters is known as parameterisation and can significantly influence performance, as demonstrated in [36], where several LDBP models employing varying parameterisations were compared. The *linear steps* can be fully implemented in either the time or frequency domain. When employing frequency-domain linear steps, linear stages are either left unoptimised [36] or the dispersion parameter β_2 in the required transfer functions becomes the primary trainable

parameter [69]. In contrast, time-domain implementation allows for a larger number of optimisable parameters, as the taps of time-domain filters can be individually optimised. For the *nonlinear steps*, the commonly optimised parameter is the nonlinear parameter γ in the function $\boldsymbol{\sigma}$. Alternatively, employing an "enhanced" approach [117] involves filtering the power waveform with trainable FIR filters, providing additional trainable parameters in the nonlinear steps. Among these various configurations, employing time-domain filters in both the linear and nonlinear steps and allowing all filter taps to be trainable has demonstrated the best performance [36].

LDBP was initially developed for single-channel equalisation, with numerical studies assessing its performance on single-channel and WDM transmission scenarios [66, 58]. These studies demonstrated improved performance and reduced computational complexity compared to conventional DBP. In transmission systems with practical baud-rates, the performance gains over DBP at the same number of steps were as high as 2.1 dB in effective SNR [58]. In terms of complexity, it was shown that the LDBP scheme could cost as little as 3.5 times the complexity of linear equalisation due to low filter length requirements [58]. Experimental validations of the scheme soon followed, with researchers investigating the integration of the scheme into practical systems. In single-channel experiments, LDBP showed substantial gain with respect to DBP at the same number of steps, and achieved a 1.9 dB improvement over CD compensation at six times less complexity than conventional DBP [9]. However, in WDM transmission, the scheme offered limited improvements (a maximum of 1 dB Q-factor gain over CDE for a 5-channel system) [34]. Another multichannel experiment showed only 0.3 dB SNR improvement over CDE, despite requiring a significantly larger number of steps per span (StpS) than numerical studies suggested [125].

These limited improvements prompted the exploration of LDBP for multichannel equalisation. The success of the LDBP approach in reducing the complexity of single-channel equalisation further motivated its consideration for this case, where the computational burdens of conventional DBP are significantly higher. Initially, time-domain MIMO LDBP equalisers were proposed for wideband single-channel signals and WDM systems, featuring fully trainable steps implemented using convolutional layers [124, 67]. Later, learned MIMO LDBP equalisers relying on frequency-domain filtering were also developed for WDM transmission [69], using the architecture in [94] as a starting point. The latter approach focused on making model training more practical, achieving significant training cost reductions by

reducing the number of trainable parameters and explicitly deriving the required gradients. Leveraging filtering in the nonlinear steps enabled single-step per span implementations that delivered significant performance gains. Compared to conventional schemes, these MIMO LDBP models demonstrated a better tradeoff between the number of processed channels and computational cost. While in DBP processing, providing effective equalisation to a large number of channels required more computational steps, the number of steps needed by LDBP to demonstrate improvement were much lower. Initial numerical demonstrations showed favourable results. In an 11-channel WDM transmission, a 5-channel LDBP model provided 1.2 dB improvement in Q^2-factor [124]. Additionally, the scheme achieved a 0.75 dB greater gain than conventional MIMO DBP when both models maintained equal complexity. However, the gains reported in experimental validations have been moderate. For instance, the experiment in [69] showed an improvement similar to two-step per span MIMO DBP, while requiring a complexity similar to single-channel DBP at two step-per-span.

In addition to the limited improvements demonstrated in experiments, LDBP has several other limitations. It has been shown that DBP architectures may introduce spectral artefacts that induce out-of-band distortions and reduce performance [25]. Furthermore, analysis of the optimized parameters of a learned DBP algorithm has revealed that, for the scheme to operate effectively, it must not only invert the fibre channel but also mitigate distortions that are self-generated or exacerbated by the DBP process itself [34]. Figure 2.11 depicts the magnitude spectra of the learned filters along a learned DBP scheme, reflecting the findings reported in [34]. Filters from later stages display M-shaped magnitude spectra with high-pass characteristics, in contrast with the pass-band response of ideal CD filters

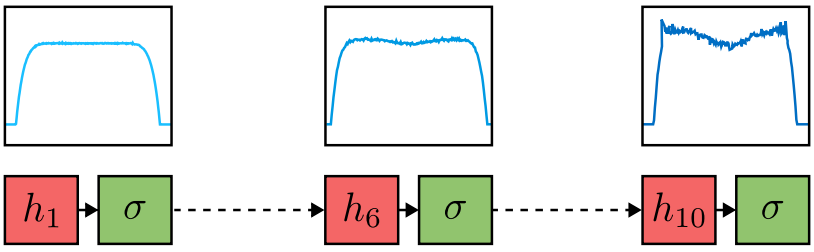


Figure 2.11: Illustration of the findings in [34], depicting the evolution of the spectra of linear operators of a learned DBP algorithm. The figure shows steps 1, 6 and 10 of an LDBP algorithm of 10 stages. An artistic representation of the amplitude spectrum of the each algorithm step is shown. The amplitude spectra are "M" shaped, suggesting that the filters not only compensate for dispersion, but also correct for limitations of the architecture by mitigating distortions caused by uncorrected phase rotations, which accumulate throughout the sequential structure.

used in conventional DBP. Assuming a received signal with field E_z, this behaviour can be explained by the presence of a residual distortion term $|E_z|^2 E_z$, which has a ∩-shaped spectrum. As it accumulates along the structure due to imperfect phase rotations, the M-shaped filters try to supress it. This distortion term spans a much larger bandwidth than E_z, leading to out-of-band distortion. A non-sequential architecture could avoid the accumulation of self-induced distortion and offer improved equalisation effectiveness. Additionally, integrating CD compensation into a nonlinear equalizer with a sequential architecture may be suboptimal. CD equalisation is a critical operation that must be performed reliably, and performing it with filters optimised through a data-aided approach that prioritises nonlinear equalisation may compromise its effectiveness. Finally, the extensive sequence of filtering steps or "layers" introduces significant latency. While LDBP reduces latency compared to DBP by lowering the number and complexity of steps, proposed implementations still require at least one step per fibre span [1], limiting further latency reductions [9].

An Alternative to LDBP

The success of LDBP has demonstrated the potential of model-driven approaches, paving the way for developing other schemes where supervised learning addresses the shortcomings of conventional equalisers. In this thesis, we explore using the same gradient-based approach applied in LDBP to Volterra-based algorithms. While LDBP was motivated by structural similarities between DBP and deep neural networks, we propose that the successful optimisation of conventional algorithms does not require them to have the same functional form as a deep neural network. The Volterra architecture we have selected is the IVSTF (or first-order RP) model [88, 136]. While this algorithm does not resemble a deep neural network, its architecture resembles the interconnection of **single-layer perceptrons**, as shown in Fig. 2.12. Each branch of the IVSTF consists of a short sequence of operations—linear-nonlinear-linear, while the single-layer perceptron comprises a linear layer followed by a nonlinear activation. These similarities raise the possibility of modifying the IVSTF to achieve the adaptability that characterises neural networks.

Despite the similarities between the IVSTF and single-layer perceptrons, key differences exist, which could hinder the successful use of the IVSTF as a trainable scheme. While the

[1] To date, *learned* DBP implementations operating with fewer than 1 step per span, such as DBP methods based on the "logarithmic-perturbation" method [20], have not been introduced.

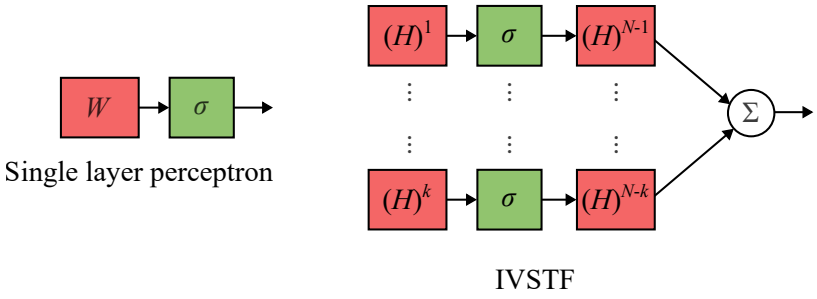
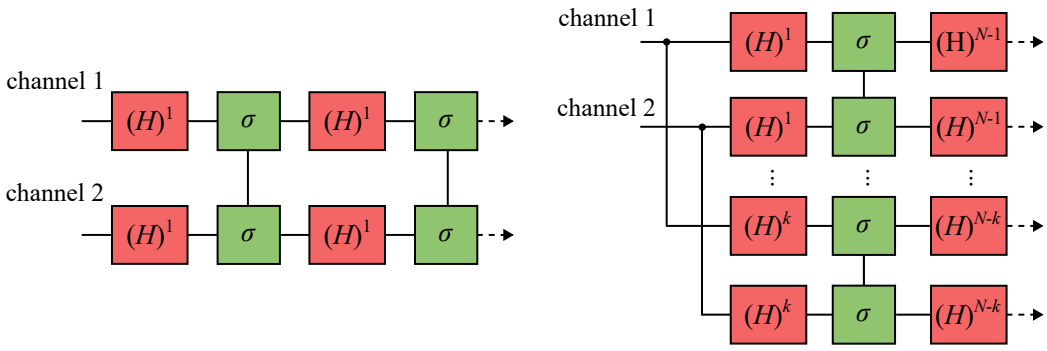


Figure 2.12: Comparison between the functional forms of IVSTF branches and a single layer perceptron. In the IVSTF diagram, $(H)^k$ represents a CD transfer function H addressing k fibre spans, while σ represents a nonlinear activation function.

perceptron's layer is fully connected, represented by a two-dimensional weight matrix W, the IVSTF's linear step uses simpler a one-dimensional filter, potentially limiting adaptability. Furthermore, unlike the perceptron, IVSTF branches implement their linear steps in the frequency domain, deviating from the architecture of highly adaptive NN models. While the model's frequency domain linear steps enable low computational complexity and optimised data block processing, they offer fewer viable trainable parameters compared to time domain methods [69]. However, the branches —and by extension, the entire IVSTF— employ easily differentiable operations, and consequently can be trained in the same way as the perceptron. To address the reduced dimensionality of the linear steps, time-domain parallel implementations could be pursued to enhance learning, potentially leading to learning capabilities similar to time-domain LDBP models.

Optimising the IVSTF using ML methods could extend its applicability to multichannel scenarios. Historically, accuracy limitations stemming from the third-order truncation to the Volterra series have restricted its use, leaving DBP as the primary physics-based option for MIMO operation. Gradient-based optimisation may address the inaccuracy of the truncated nonlinear phase shift approximation, enabling precise estimation of interchannel impairments. To extend single-channel IVSTF to MIMO approaches, the approach to modelling interchannel interactions employed in MIMO LDBP algorithms could be followed. Specifically, the nonlinear stages of MIMO DBP, given by the coupled NLSE SSF solution and which efficiently accounts for adjacent channel interactions in learned models [124, 69], could inform the adaptation of the IVSTF. Furthermore, the parallelism inherent to the IVSTF may be particularly beneficial in multichannel contexts. In a MIMO IVSTF realisation, the nonlinear distortions experienced by each channel and induced by



MIMO DBP MIMO IVSTF

Figure 2.13: Cascaded and parallel architectures of 2-channel MIMO models implemented with 1 step per span: (a) MIMO DBP (b) MIMO IVSTF. H represents the CD transfer function of a fibre segment, while σ represents the appropriate activation function for each model.

each fibre segment would become parallel signal paths. This is shown in Figure 2.13, which contrasts a MIMO DBP architecture with a potential MIMO IVSTF design. Unlike MIMO DBP, where channel paths increase in depth with link length, signal paths in the IVSTF always keep the same short length regardless of transmission distance. This characteristic of MIMO IVSTF-based schemes could make multichannel equalisation more feasible for practical deployment.

The following section summarises the research questions addressed in this thesis.

2.3 Research Questions

- *How can domain knowledge from optical communications further contribute to the development of ML-based equalisers?*

  In the field of optical communications, the extensive understanding available on the physics of signal impairments is used to devise strategies to improve signal quality. Inverse models derived from the VSTF have shown potential for equalisation, but are hindered by imprecision resulting from truncating the Volterra series. Combining this modelling with ML optimisation could potentially result in more efficient equalisation schemes.

- *How can ML enable the design of model-driven single-channel equalisers based on parallel structures?*

  Most current approaches combining ML with domain knowledge rely on the SSF

method, which uses a sequence of linear and nonlinear steps. While these schemes offer low complexity, the sequential filtering of the signal can lead to limited effectiveness due to error propagation and increased latency. We investigate parallel architectures, which offer shorter signal paths than SSF-based solutions, potentially mitigating these issues and improving suitability for hardware implementation.

- *How can machine learning support the operation of IVSTF-based multichannel equalisers?*

  The precision limitations of the third-order IVSTF in approximating signal propagation has hindered its application in multichannel equalisation. We investigate how ML can address these limitations and enable the operation of IVSTF-based MIMO equalisers. Several architectures based on the IVSTF framework have been proposed and examined.

- *Can effective guidelines be devised for the design of model-driven multichannel equalisers?*

  Our research demonstrates that multiple model-driven architectures can be employed for multichannel equalisation. The various proposed time and frequency domain implementations have specific design parameters that must be carefully set to obtain the best performance and lowest complexity possible. Therefore, in addition to developing architectures suitable for various transmission scenarios, we aim to devise an effective parameter configuration procedure for these models.

Chapter 3

Fundamentals of Machine Learning

This section introduces the principles underlying the optimisation techniques applied in this thesis. We introduce fundamental concepts related to the design of artificial neural networks (ANNs). We give an overview of their architecture, basic components, and training process. We explain these concepts with the multi-layer perceptron (MLP), a type of feed-forward ANN. However, the concepts also apply to more advanced neural networks as well as the model-driven approaches proposed in this thesis.

3.0.1 Artificial Neural Networks

ANNs are computational models developed for pattern recognition tasks. The term "neural network" originated from studies on how biological neurons process information. Fundamentally, a neural network is a series of functional transformations. A basic neural network first takes input variables $x_1, x_2, \ldots x_N$ and uses them to compute activations a_q for $q = 1, \ldots, H$, given by

$$a_q = \sum_{p=1}^{N} \omega_{qp}^{(1)} x_p + \omega_{q0}^{(1)}, \tag{3.1}$$

where $\omega_{qp}^{(1)}$ are the weights and $\omega_{q0}^{(1)}$ the biases of the first layer. These activations are transformed by the activation function $\sigma(\cdot)$ to produce the outputs of the hidden units $z_q = \sigma(a_q)$. The layer is called a hidden layer since its units become the inputs for the

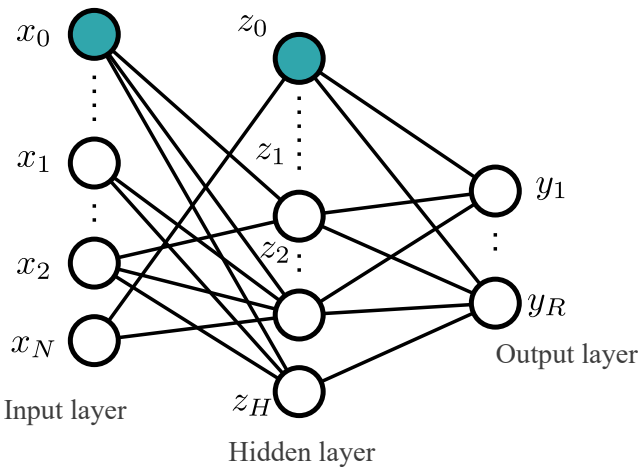


Figure 3.1: Diagram of an artificial neural network with one hidden layer. Circles represent units, and the lines between them represent the model's weights. The coloured circles indicate bias units.

second layer, where activations a_r are computed as

$$a_r = \sum_{q=1}^{H} \omega_{rq}^{(2)} z_q + \omega_{r0}^{(2)}. \tag{3.2}$$

Here, $r = 1, \ldots, R$ is the number of output units of the second layer. The process is repeated until the output layer is reached. If the network has only one hidden layer, the second layer becomes the output layer. A network with more than one hidden layer is called a deep neural network (DNN). Since using more layers enhances the model's generalisation capability [53], DNNs have become ubiquitous in modern applications. The choice of activation functions is crucial and depends on the type of data to be processed and the predictive task the network will be used for. For regression problems, the activation functions for computing the outputs are the identity, resulting in $y_r = a_r$. The structure described above constitutes a fully connected network. However, implementing a sparse network in which some weights are equal to zero is also possible, eliminating connections between a layer's inputs and outputs.

Activation Functions

In ANNs, the activation functions are continuous functions that introduce nonlinearity to the model, enabling it to approximate nonlinear behaviour. Without these, a multi-layer perceptron would be equivalent to a single linear layer and could only approximate linear

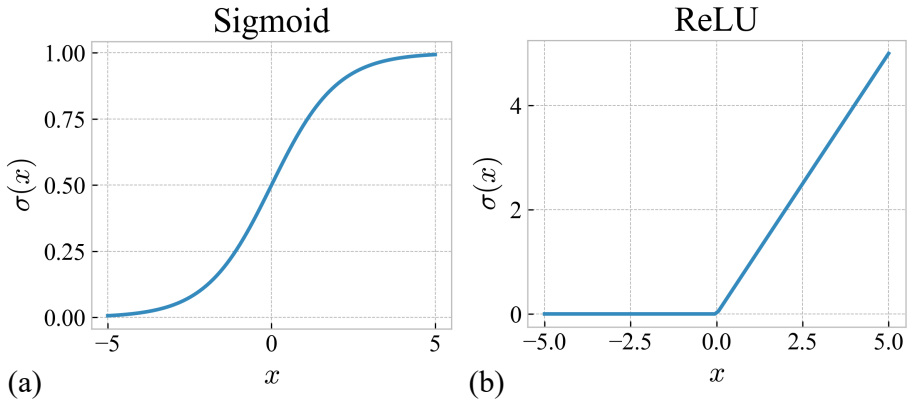


Figure 3.2: Commonly used activation functions: (a) sigmoid, which maps inputs to values between 0 and 1 with a smooth transition, (b) rectified linear unit (ReLU), which outputs zero for negative inputs and grows linearly for positive inputs.

functions. A popular choice for the nonlinear activations of ANNs is the sigmoid function:

$$\sigma(x) = \frac{1}{(1 + exp(-x))}. \tag{3.3}$$

Other functions are also ubiquitous, such as the ReLU [52] and the hyperbolic tangent (tanh). The shape of the sigmoid and ReLU functions is shown in Figure 3.2.

In ML models following the model-driven paradigm explored in this thesis, custom activation functions derived from a physics model may be employed [41]. An example is LDBP, where the activation function $f(x) = xe^{j\gamma c|x|^2}$ is taken from the nonlinear step of the SSF method [9]. Similarly, trainable models based on the Volterra series involve nonlinear steps that apply point-wise functions $f(x) = j\gamma c|x|^2 x$, which is a first-order polynomial approximation of the exponential function from DBP. These functions are complex-valued, and their behaviour may be analysed with a representation in the complex plane. Figure 3.3 shows the magnitude and phase of these nonlinear functions.

3.0.2 Neural Network Training

The optimisation of the parameters of a neural network often relies on a set of data samples, referred to as training data. ANNs are widely used in supervised learning tasks, where the training data consists of input vectors $\mathbf{x}_n$ with $n = 1, ..., N$ with corresponding target vectors $\boldsymbol{t}_n$. Determining the neural network's parameters, a process known as training, is an optimisation problem involving minimising an error or **loss function**. The process is

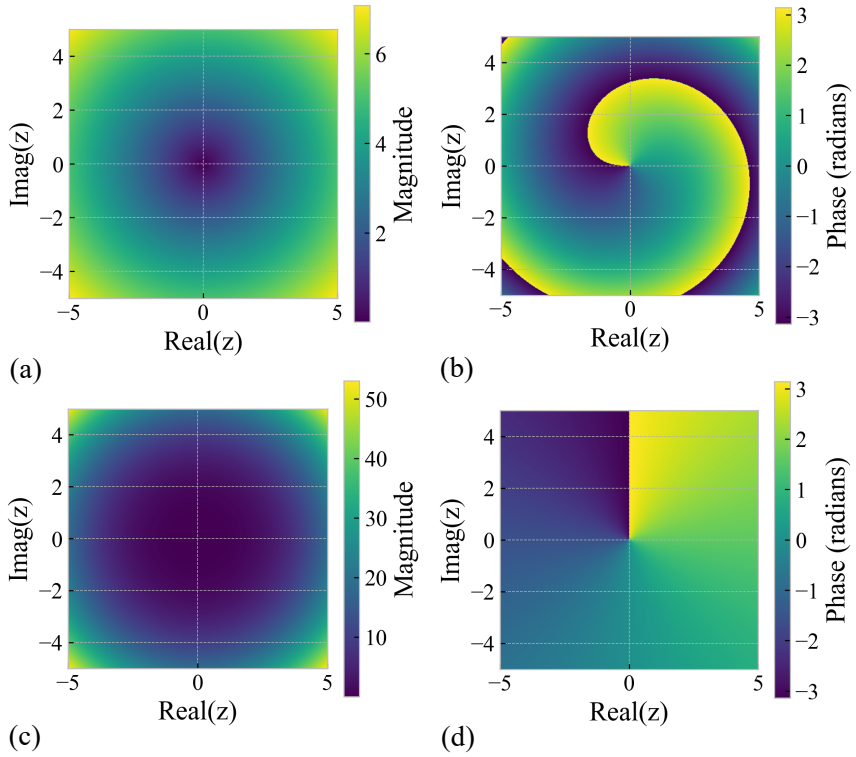


Figure 3.3: Magnitude and phase of the complex-valued activations employed in the nonlinear steps of (a, b) DBP: $f(x) = xe^{jc|x|^2}$ and (c, d) Inverse Volterra series transfer function: $f(x) = xjc|x|^2$.

iterative, comprised of steps in which the gradient of the loss function with respect to each of the networks parameter is calculated, and then employed to update the weights of the model. The loss function compares the output of the neural network $\{\mathbf{y}(\mathbf{x}_n, \mathbf{w})\}$ with target vectors $\{\boldsymbol{t}_n\}$. The appropriate loss function depends on the activation function of the output units, chosen according to the predictive task the NN is employed for. For regression tasks, the output units are linear and the most commonly used error function is least-squares. For classification, the cross-entropy error function is employed, and the outputs are logistic sigmoid for the binary case and softmax for the multiclass case. The prevalence of mean squared error functions and cross-entropy error functions is largely due to the simplicity in calculating their gradients, supporting training convergence.

The mean squared error loss function is given by

$$E(\mathbf{w}) = \frac{1}{2}\sum_{c=1}^{N} \|y(\mathbf{x}_n, \mathbf{w}) - \mathbf{t}_n\|^2. \tag{3.4}$$

The weight vector $\mathbf{w}$ of the neural network defines a parameter space over which the error

function is defined. Minimising the error function is analogous to finding the lowest point on an error surface $E(\mathbf{w})$, or equivalently, finding the parameters $\mathbf{w}$ where $\nabla E(\mathbf{w}) = 0$. In practice the error function is non-convex, often leading the optimisation process to find a local minima rather than a global solution. Nevertheless, convergence to a local minimum can still result in successful training outcomes for the neural network.

Solving the equation $\nabla E(\mathbf{w}) = 0$ requires iterative numerical methods. The most common approach is to give an initial value to the weight vector $\mathbf{w}^{(0)}$ and then update it iteratively. Many algorithms for computing the weight update utilise gradient information, requiring the evaluation of $E(\mathbf{w})$ at each new weight vector. The most basic approach updates the weights by taking a step in the direction opposite to the gradient. A widely used algorithm is stochastic gradient descent (SGD), which updates the weight vector $\mathbf{w}$ on a per-sample basis [8]:

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E(\mathbf{w}^{(\tau)}), \tag{3.5}$$

where η is the **learning rate** and τ is the step. In this approach, data can either be processed sequentially or randomly. The learning rate determines how large will the update to the networks parameters be. Careful tuning of the learning rate is essential, as a small value may slow down training, while a large value risks causing training divergence. Adjusting how input data is fed to the neural network gives rise to different variants of gradient descent. A common approach is dividing it into sets called mini-batches. A suitable algorithm for this case, called **mini-batch gradient descent**, updates the weights on a per-batch basis. In mini-batch training, a pass through the entire dataset is called an **epoch**, a hyperparameter employed to define training duration. The **batch size** is an optimisable parameter. A large batch size improves the accuracy of the parameter updates, as a larger batch of training data provides more information per training step. In contrast, a small batch size reduces the information intake per step, which can cause the steps to deviate more from the path to the minimum of the cost function. In practice, the batch size is limited by the available computer memory resources.

Mini-batch gradient descent is sensitive to the asymmetry of the cost function with respect to weight updates. Variants such as the Adam optimiser address this issue by incorporating momentum terms based on previous updates [76]. The Adam optimiser is widely

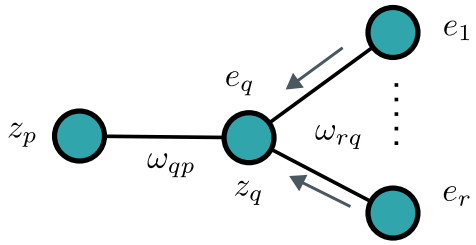


Figure 3.4: Representation of the backpropagation method. The arrows indicate how errors e are propagated backwards through the network.

used for its efficiency, often requiring fewer iterations to converge than other algorithms. All models in this thesis were trained using Adam. We expect this algorithm to be particularly advantageous in model-driven schemes, where parameter updates involve diverse types (e.g. real and complex values) and magnitudes.

The gradient ∇E can be efficiently computed using the backpropagation technique, which propagates information backwards through the model to evaluate the required derivatives. The method is not limited to the multilayer perceptron and can be applied to other architectures [8]. Backpropagation involves two main stages: A forward pass and a backward pass. In the forward pass, an input vector is propagated forward through the network, and the activations of all hidden and output units are computed. In the backward pass, the chain rule for partial derivatives is applied to calculate the derivative of the loss function with respect to each weight. To formally describe this process, we define the summed input to a unit p as $a_p = \sum_q \omega_{pq} z_q$, where z_q is the activation of unit q. The activation for unit p is then given by $z_p = \sigma(a_p)$. Using the chain rule, the derivative of the error function with respect to a weight ω_{pq} for a given input pattern n is expressed as

$$\frac{\partial E_n}{\partial \omega_{pq}} = \frac{\partial E_n}{\partial a_p} \frac{\partial a_p}{\partial \omega_{pq}}. \tag{3.6}$$

For convenience, the error associated with a unit p is defined as $e_p = \frac{\partial E_n}{\partial a_p}$. For the output units, the errors can be obtained simply as $e_r = y_r - t_r$, where y_r is the output calculated during the forward pass and t_r is the target value for unit r. For the hidden units, the backpropagation formula [8] allows us to compute the errors by "propagating" errors backwards from deeper layers in the network as follows:

$$e_q = \sigma'(a_q) \sum_r \omega_{rq} e_r. \tag{3.7}$$

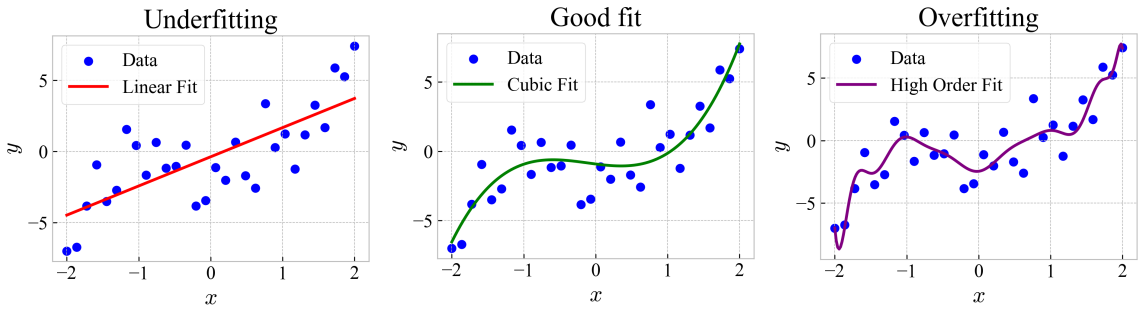


Figure 3.5: Illustration of data fitting outcomes: (a) underfitting, (b) good (balanced) fit, (c) overfitting.

This formula is applied recursively for all hidden units. Figure 3.4 represents the propagation of errors through the units of a network that has three layers.

Generalisation

The effectiveness in training a neural network is determined by its ability to perform a specific task on unseen data. The performance achieved on the training set alone does not reliably indicate how well will the model generalise. This is due to the problem of **overfitting**, where the model learns patterns specific to the training data, leading to poor generalisation. Stopping training prematurely to avoid this issue, or using a model incapable of capturing the relevant patterns the task requires, can lead to **underfitting**. To obtain intuition on these phenomena, it is instructive to consider simple polynomial models. Figure 3.5 illustrates various polynomial fits of different orders on data that follows a cubic distribution and is affected by Gaussian noise. In this case, it is desirable to learn the cubic distribution, not the noise. Employing a high-order polynomial results in overfitting, where both the desired data pattern and the noise are captured. Conversely, employing linear regression leads to underfitting.

To find the model that generalise best, a common approach is to train multiple models which are then compared on a **validation set**. Since overfitting to the validation data could occur under certain conditions, a **test set** is employed to evaluate the final performance. To prevent a model from overfitting to the training data, the main strategies are increasing the amount of training data and the use of regularisation techniques. **Early stopping** is a common method used to terminate a model's training once its accuracy has stopped

improving. In its most basic implementation, the number of consecutive iterations without improvement is monitored during training. Training is halted if this number exceeds a threshold called the "patience" parameter, which varies depending on the model being optimised.

Hyperparameter Tuning

Defining the architecture of neural networks is a crucial step for their effective implementation. This architecture is defined by "hyperparameters", including the number of layers, the number of units per layer, and the type of activation functions used. The choice of hyperparameters is considered one of the bottlenecks in designing NN models since it is a resource-intensive process. There is an absence of analytical rules to find optimal hyperparameters, which must be found through iterative search strategies. Several techniques have been proposed whose efficiency depends on the model's architecture. **Grid search** involves sweeping the value of each hyperparameter over a predefined range, giving all hyperparameter values the same importance. In contrast, **random search** samples hyperparameter values from an assumed probabilistic distribution. This method can outperform grid search in cases where the model's performance is mostly determined by a small number of hyperparameters [7]. The bayesian optimisation approach has recently become popular due to its efficiency, outperforming random search [135]. It uses a probabilistic model of the relationship between hyperparameters and the ML model's optimisation objective, selecting a set of promising hyperparameters at each iteration.

For model-driven schemes [61], the choice of architectural hyperparameters (such as the activation functions and the number of layers) is usually clear. However, precise tuning hyperparameters related to the training process can be achieved with simple techniques such as a grid or a random search. Bayesian optimisation could also be used. However, from a software standpoint, this requires custom implementations of BO that can handle custom layers, which are required for implementing model-driven schemes. In this thesis, initial tuning of some training-related hyperparameters for the proposed models was conducted using simple grid searches. These early experiments provided the required expertise to tune the hyperparameters manually in subsequent efforts.

Initialisation

The initialisation of deep learning models is crucial for their training. For NNs, an adequate initialisation helps avoid gradient issues that can hinder learning. Poor initialisation may lead to vanishing gradients, where gradients are too small to affect the learning direction, or exploding gradients, where excessively large gradient values cause the model optimisation to diverge. A common practice is to give weights initial values taken from a probability distribution. Initialisation is also important in model-driven ML schemes, where initialisation strategies have been shown to ensure fast optimisation convergence [58].

Unlike data-driven models, initialisation in model-driven schemes is often informed by domain knowledge. Initial parameters that lead to optimal convergence can be estimated using the underlying physics of the model. Examples include the initialisation of the linear steps of LDBP with CD FIR filters [124] and of the nonlinear steps with coefficients derived from perturbation analysis [87]. Relying on transmission parameters for initialisation is potentially disadvantageous, as links may operate under varying conditions throughout their lifetime. Mismatches between the transmission system and the model architecture are expected to be common. Nonetheless, model-driven approaches capable of handling imprecise initialisation may be developed. It has been shown that single-channel LDBP can converge when initialised with random values [58], although with suboptimal performance. Other equalisation architectures may mitigate the impact of parameter mismatch and lead to improved convergence.

3.0.3 Convolutional Neural Networks

The multilayer perceptrons presented before are best suited for unstructured data. In certain applications, predictions from a model should remain unchanged after certain transformations of the input variables. Although neural networks can learn this "invariance" given sufficient training data, another approach is incorporating these invariance properties into the NN architecture. This is the approach underlying CNNs, a type of feed-forward network originating from the "neocognitron" introduced in [48]. Their architecture was influenced by developments in neurophysiology, and consists of a sequence of layers that extract features from data using optimisable filters. Authors in [83] demonstrated the gradient-based training of CNNs for digit recognition, which revolutionised the field of computer vision. The

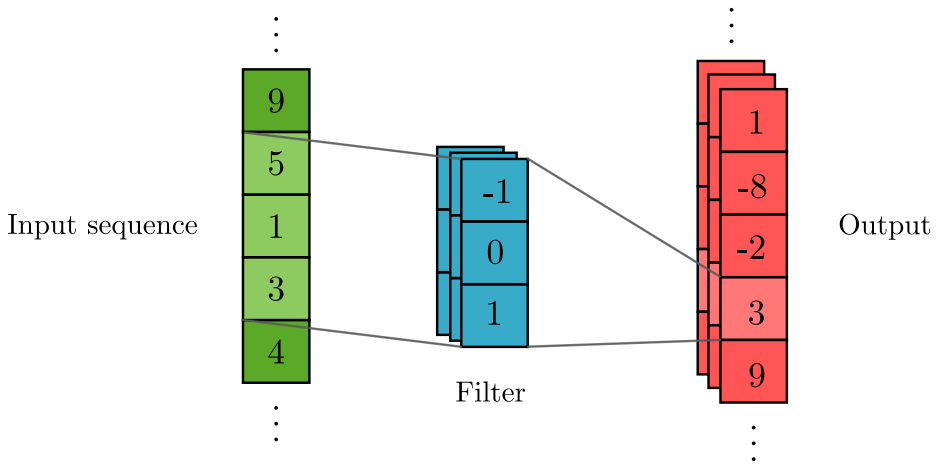


Figure 3.6: Illustration of the operation of a conv1D layer.

fundamental building block of CNNs is the convolutional layer. Due to their versatility and pattern extraction capabilities, convolutional layers are ubiquitous in deep learning models. Two-dimensional convolutional layer (conv2D) layers are suitable for computer vision applications, while one-dimensional convolutional layer (conv1D) layers have demonstrated utility in processing sequential data [77]. Convolutional layers are widely used in the implementation of NN-based equalisers. They may be used in fully convolutional architectures [84] or in combination with recurrent and dense layers.

The operation of a conv1D layer can be described as follows:

$$y_i^f = \sigma\left(\sum_{n=1}^{L}\sum_{j=1}^{K} x_{i+j-1,n} \cdot k_{j,n}^f + b^f\right). \tag{3.8}$$

Here, y_i^f is the i-th output of the convolutional layer, resulting from applying the filter f to the i-th input element, x is an input vector of length L, $k_{j,n}^f$ is the filter kernel, b^f is the bias vector and σ represents a nonlinear activation. The padding, stride, and dilation parameters further configure the layer. In the above equation, padding equals 0, and stride and dilation are equal to 1.

The operation of a conv1D is illustrated in Fig. 3.6. The layer "slides" a linear filter along the length of the input sequence. Note that the convolutional layer does *not* perform the convolution operation as defined in digital signal processing but instead does cross-correlation, which is more general. The convolution operation can be obtained by simply "flipping" the filters before passing them to the layer. Additionally, the circular convolution

can be obtained by extending the input sequence. This versatility of the layers enables them to implement any time-domain filtering operation for digital signal processing in communication systems. This has advanced model-driven approaches in machine learning-based equalisation, where conv1D layers are employed to implement trainable models with the architecture of model-based techniques (DBP, Volterra, perturbation), even enabling the implementation of all DSP stages as trainable layers [36]. This thesis extensively uses convolutional layers to implement Learned Volterra equalisers.

Chapter 4

Methodology

Developing the models presented in this thesis required building optical transmission simulators and digital signal processing implementations. This chapter details the optical transmission system models and the applied signal processing techniques.

4.1 Modelling of the Optical Transmission System

The simulated transmitter and optical channel are represented in Fig. 4.1. Pseudorandom symbol indices were generated using NumPy's `np.random.randint` function. This function uses a global instance of `np.random.RandomState`, which is based on the Mersenne Twister algorithm with a period of $2^{19937} - 1$ [95]. The indices were mapped to a constellation of QAM symbols using Gray mapping, shown in Fig. 4.2 for a 16-QAM constellation. In our simulations we have considered two different QAM orders: 16-QAM (Chapter 5) and 64-QAM (Chapters 5 and 6). However, this thesis examines transmission with 64-QAM modulation in more detail since this format imposes higher SNR requirements. All simulations consider a symbol rate of 32 Gbaud.

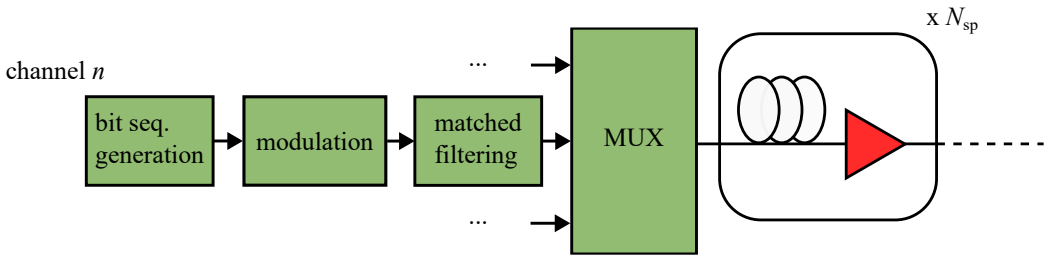


Figure 4.1: Block diagram of the Tx architecture and optical link employed in our simulations.

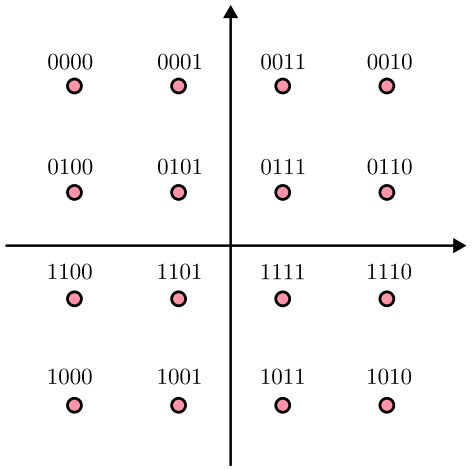


Figure 4.2: Gray mapping for the 16-QAM modulation format.

In the wavelength division multiplexing scenario, we consider a multi-wavelength signal of $2K + 1$ modulated channels with the frequency spacing $\Delta\omega$, centred at a wavelength of 1550 nm. K was a design parameter: We considered 5-channel transmission in our initial efforts, as reflected in our initial conference contribution on multichannel equalisation [13], and later 11-channel transmission to explore more realistic scenarios. The channel spacing is set to 40 GHz, placing the system in the dense WDM category. Standard WDM systems typically use 50 GHz spacing, which reduces the impact of XPM impairments and improves system performance. The narrower 40 GHz spacing is used deliberately to accentuate the inter-channel impairments and enable evaluation of equalisation gains in more tightly spaced scenarios such as super-channel systems.

Transceiver nonlinearity is a major limiting factor in optical coherent systems [131]. The transmitter laser induces phase noise, and modulators exhibit nonlinear responses which becomes important when transmitting using high order modulation formats and very high baud rates, like it is done in metro and data-centre networks. In this thesis we consider **transmission** links, where baud-rates and the order of modulation formats are not as high, and fibre nonlinearity is the primary source of impairments. Therefore, we consider transceiver impairments to be negligible, assuming an ideal conversion from the digital domain to the optical domain, and components such as lasers, modulators and analog-to-digital converter (ADC)s are omitted from the model.

The transmission links consist of interconnected SSMF fibre spans of equal length, terminated with erbium-doped fibre amplifiers (EDFA) amplifiers. The EDFAs are modelled

as amplifiers that apply a gain of αL_{sp} dB to the signal and introduce ASE noise. The ASE noise is modelled as additive white Gaussian noise (AWGN), with a spectral density given by [33]

$$G_{\text{ASE}}^{\text{EDFA}} = (e^{\alpha L_{\text{sp}}} - 1) h \nu_s n_{\text{sp}}. \tag{4.1}$$

Here, h is Planck's constant, ν_s is the operating frequency, and n_{sp} is the amplifier spontaneous emission factor. We employ fibre span lengths L_{sp} of 100 km, while the overall link lengths we consider are 600 km and 1000 km. In our simulator, signal propagation is modelled using the SSF method to solve the NLSE for single polarisation transmission:

$$\frac{\partial A(z,t)}{\partial z} = -\frac{\alpha}{2} A(z,t) + j\frac{\beta_2}{2}\frac{\partial^2 A(z,t)}{\partial t^2} - j\gamma |A(z,t)|^2 A(z,t). \tag{4.2}$$

When considering WDM transmission, $A(z=0,t) \triangleq \sum_{k=-K}^{K} A_k(t) e^{jk\Delta\omega t}$, where $A_k(t)$ is the complex field envelope of each wavelength channel. At the end of the link the received signal $A(L,t)$ is de-multiplexed and each sub-channel is detected coherently. Although this setup excludes PMD— a significant impairment in current optical networks [140]—we do not anticipate this omission to compromise the study's conclusions on the potential gains from the equalisation of Kerr-based impairments. This is because PMD is a linear impairment weaker than chromatic dispersion, which can be mitigated by adaptive algorithms to an extent in modern coherent receivers [71]. Extending our schemes for polarisation multiplexed systems is left for future work. The symmetric variant of the SSF method [37], which localises the nonlinear phase shift at the centre of the span, has been employed for its accuracy. Therefore, each step in the method consists of three substeps: an initial linear step covering a length $h/2$, a nonlinear step addressing the nonlinear phase shift of h, and a second linear step identical to the first. A small step size — or, equivalently, a large number of steps — ensures that linear and nonlinear effects can be treated separately with sufficient precision. We implemented an adaptive step approach that adjusts the number of steps based on changes in the nonlinear phase. The required number of steps depends on the signal bandwidth: An increase in the number of WDM channels increases the step requirement.

An ideal optical front-end is assumed at the receiver, with perfect conversion from the

Figure 4.3: Block diagram of the single-channel receiver DSP architecture.

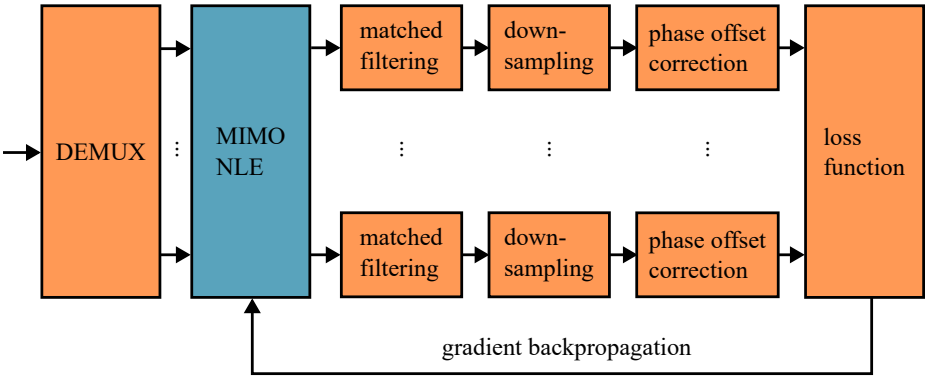


Figure 4.4: Block diagram of the MIMO receiver DSP architecture.

optical to the digital domain. Front-end components, including the optical hybrid and photodetectors, are not modelled since they are assumed to perform without signal loss or nonlinear distortion. Each channel is shifted to the baseband. Then, a sinc filter is applied to each channel. The signals are then downsampled to two samples per symbol, which is sufficient for DBP and Volterra-based single-channel and MIMO equalisers.

The computations involved in the simulation of the optical link have been implemented in the Tensorflow 2 framework to leverage graphics processing unit (GPU) acceleration. The initial version of our simulator (employed for the results in [12]) independently generated, transmitted and received short data batches, similar to the implementation in [57], allowing native block processing and facilitating training routines at the receiver. This approach provided sufficient accuracy for single-channel scenarios with relatively low channel memory. However, to accurately account for the impact of propagation on the broader signal bandwidths in multichannel transmission, a simulator capable of processing large sequences was developed, along with the necessary overlap-based block processing at the receiver.

4.2 Digital Signal Processing

We have adopted the simplified DSP architecture shown in Fig. 4.3. The receiver DSP is comprised by NLE, matched filtering, downsampling and phase offset correction. A standalone CD compensation stage is omitted when using Volterra-based NLEs since CD

equalisation is already integrated in them. Moreover, adaptive equalisation is not required, as our simulated system does not account for transceiver impairments or volatility of the transmitted channels. The architecture in Fig. 4.4 is employed when applying MIMO equalisation. In this case, a subset $2M + 1$ ($M \leq K$) of the received sub-channels is processed, while the remaining sub-channels are discarded. Let $\boldsymbol{r}$ be the equalised signal of a given channel. After equalisation, downsampling is performed by retaining only one sample of each symbol interval, along with normalisation to account for the channel power and the effect of downsampling:

$$\tilde{\boldsymbol{s}}[m] = \frac{\boldsymbol{r}[m \cdot \mathrm{OS}_d]}{\sqrt{P_{\mathrm{ch,lin}}} \cdot \sqrt{\mathrm{OS}_d}}. \tag{4.3}$$

Here, m represents the indices of the samples in the processed block, and OS_d is the digital oversampling rate. Subsequently, phase offset correction is applied according to [58]:

$$\hat{\boldsymbol{s}} = \tilde{\boldsymbol{s}} e^{-j\hat{\phi}}, \tag{4.4}$$

where $\hat{\phi} = -\arctan\left(\frac{\mathrm{Re}(q)}{\mathrm{Im}(q)}\right)$, $q = \boldsymbol{s}^* \cdot \tilde{\boldsymbol{s}}$ and $\boldsymbol{s}^*$ denotes the conjugate of the transmitted symbol vector $\boldsymbol{s}$. This phase correction is genie-aided, since it assumes the prior knowledge of $\boldsymbol{s}$. Symbol demodulation is performed using a hard-decision nearest-neighbor algorithm. Finally, decimal integers are converted to binary representations for bit error counting. During the training of NLE algorithms, BER calculation is omitted and the output signal is employed to compute a loss from which parameter updates can be calculated. To enable the training of the NLE schemes, which requires the backpropagation of gradients through the DSP chain, all of the DSP operations after the NLE and before the BER calculation are implemented as layers in the Tensorflow framework.

In the following sections we describe the linear filtering techniques employed in the implementation of our equalisation schemes.

4.2.1 Linear Filtering

While a dedicated CD compensation stage is not a part of our DSP architecture, CD filtering is extensively used in the models developed in this thesis. The dispersive effects introduced by standard single mode fibre can be described by a discrete linear time invariant (LTI)

model and characterised in the frequency domain by the following **transfer function**

$$H_{\mathrm{CD}}(\omega) = e^{-jM(\omega T)^2}, \tag{4.5}$$

where $M = \frac{D\lambda^2 L}{2cT^2}$ and D is the dispersion parameter, λ is the transmission wavelength, L is the fibre length, c is the speed of light and T is the sample period. This transfer function constitutes an all-pass filter which effects can be compensated with the transfer function

$$H_{\mathrm{CD}}^{-1}(\omega) = (H_{\mathrm{CD}}(\omega))^{-1} = e^{jM(\omega T)^2}. \tag{4.6}$$

Performing this filtering involves an element-wise multiplication between the frequency representation of the signal and the transfer function. Since the signals in our study are finite temporal sequences of N equally-spaced samples, the appropriate method for obtaining their frequency-domain representation is the **DFT**, which converts a sequence $\{x(n)\}$ of length $L \leq N$ into a sequence $\{X(k)\}$ of length N. The k^{th} element of the signal's DFT is given by [106]

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j2\pi \frac{k}{N} n}. \tag{4.7}$$

FFT algorithms can efficiently compute the DFT. These algorithms are extensively used in this thesis to efficiently "switch" the processing between time and frequency domains.
In WDM systems, wavelength channels co-propagating over an optical fibre travel at varying velocities, resulting in delays between them. The walk-off of channel i relative to the central wavelength after propagation over a fibre length L can be defined as [67]

$$D_i(\omega) \triangleq e^{-j\beta_2 L\omega\Delta\omega_i}, \tag{4.8}$$

where $\beta_2 = -D \cdot \lambda^2/(2\pi c_0)$ and $\Delta\omega_i$ is the frequency offset from the central frequency for channel i. In MIMO processing, the precise correction of this delay is essential to achieve interchannel impairment compensation [67, 124]. The walk-off delay can be easily compensated in the frequency domain, and included in the inverse linear transfer functions

employed in nonlinear equalisers [69]. For channel i and a length of fibre L,

$$H_i^{-1}(\omega) = (H_{\mathrm{CD}} D_i(\omega))^{-1} = e^{j\beta_2 L(\omega^2 + 2\omega\Delta\omega_i)/2}. \tag{4.9}$$

To enable efficient processing, transceivers in optical transmission systems process data sequences in blocks. When implementing chromatic dispersion compensation in the frequency domain within block-based DSP —an essential operation within Volterra and DBP equalisers— a key challenge arises: intersymbol interference occuring at the block edges needs to be addressed. Furthermore, since frequency domain filtering is carried out using FFT operations, the equivalent cyclic convolution applied to each block introduces time aliasing [21]. Overlap-based methods are commonly used to address these issues. We have implemented the overlap-and-save [142] method in our DSP pipeline to support our single-channel and MIMO-based equalisation solutions. A detailed investigation of the overlap-and-save requirements for MIMO based equalisation is provided in chapter 6. In our MIMO DSP, the overlap-and-save method is applied independently to each channel. The data sequence for each channel is first extended by adding zeros to the beginning and end, allowing it to be divided into blocks of equal length $N_{\mathrm{data}} = N_{\mathrm{FFT}} - N_{\mathrm{Overlap}}$. Subsequently, overlapping is performed by "saving" N_{Overlap} samples from the block $i-1$ and appending them to the block i. The overlapped blocks, which now have each a length N_{FFT}, are then processed by the receiver DSP. Time aliasing occurs at the block edges during CD filtering. However, if N_{Overlap} is sufficiently large, only the overlap samples are affected. After processing, the correctly filtered samples are obtained by removing the overlap samples and the added zeros.

We next examine the effect of block-wise processing on the performance of frequency domain walkoff-delay correction, employed in some of our MIMO schemes. We consider the compensation of chromatic dispersion and walk-off delay for a 600 km SSMF link, performed using Eq. (4.9). To emulate different block sizes, transfer functions are defined for various lengths of the frequency vector ω. We consider $\beta_2 = -2.1683 \times 10^{-26}\,\mathrm{ps}^2/\mathrm{km}$. Figure 4.5 (a) illustrates the phase response of the transfer functions for channels centred at varying frequencies $\Delta\omega_i$. We observe that each phase response has a quadratic shape, as expected from the exponent of the transfer function in Eq. (4.9). The phase responses rotate counterclockwise about the origin as a function of the centre frequency. Figure 4.5 (b) shows the

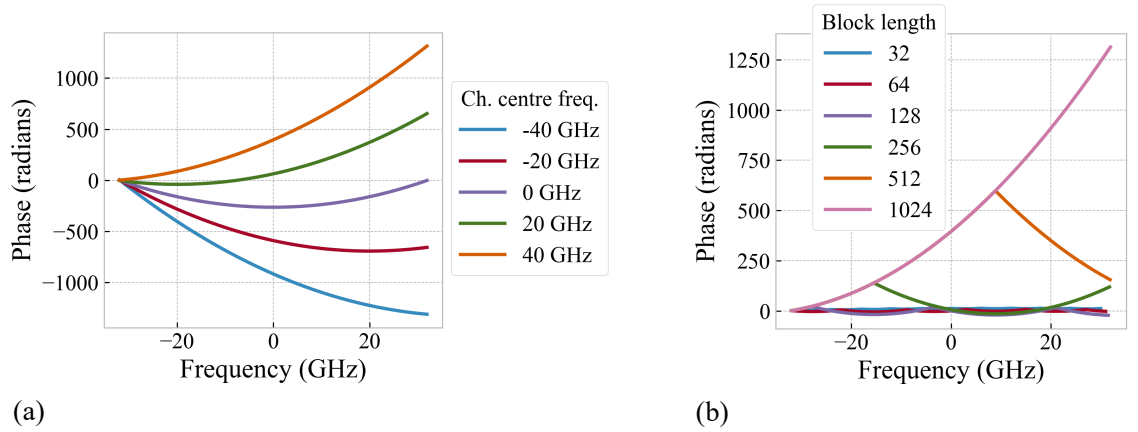


Figure 4.5: Phase response of (a) the transfer functions for compensating CD and walkoff delay of channels centered at various wavelengths (b) transfer function for a channel centered at 40 GHz when varying the block length.

phase response of the transfer function for a channel centred at $\Delta\omega_1 = 2\pi \cdot (40\,\text{GHz})$ (orange curve in Fig. (a)). A minimum block length of 1024 samples is required for accurate walk-off delay compensation, as transfer functions of shorter lengths fail to reproduce the desired phase response accurately. Finally, we note that in single-channel overlap equalisation, adequately configuring the overlap length is crucial to avoid performance penalties, as it must be large enough to prevent inter-block interference [79]. While needing to exceed the overlap, the block length is optimised primarily to reduce complexity and is commonly set to twice the overlap length. In contrast, these results indicate that in multichannel block processing, the walk-off term imposes an additional constraint on the minimum block length beyond the usual overlap considerations.

Time-domain Filtering

CD filtering can also be performed in the time domain using FIR filters. To achieve this, time-domain filters must be designed to approximate the CD frequency-domain transfer functions, for which various methods are available [113, 121]. This approach leverages the property that circular convolution in the time domain is equivalent to multiplying two N-point DFTs in the frequency domain. For discrete signals $x_1(n)$ and $x_2(n)$,

$$\text{DFT}\left[x_1(n) \circledast x_2(n)\right] = X_1(k)X_2(k), \tag{4.10}$$

where $X_1(k)$ and $X_2(k)$ are the respective N-point DFTs and $\circledast$ is the circular convolution

operator. The circular convolution is defined as

$$x_1(n) \circledast x_2(n) = \sum_{n=0}^{N-1} x_1(n) x_2((m-n))_N \quad \text{for } m = 0, 1, \ldots, N-1, \tag{4.11}$$

where the index $((m-n))_N$ indicates an N-point circular shift.

FIR CD filters are extensively used in the nonlinear equalisers presented in this thesis. While the optimal FIR filter taps employed in the equalisers are ultimately found through a gradient-based optimisation process, their initial values are computed using numerical design methods, which aim to produce a frequency-domain response that closely approximates the ideal CD transfer function within the compensation band. An early design method for FIR CD filters, known as direct sampling, was proposed in [113]. In this method, an inverse discrete Fourier transform (IDFT) is applied to Eq. (4.6) to derive the CD impulse response, which is then is truncated. The coefficients of the odd-length filter are given by

$$a_k = \sqrt{\frac{jcT^2}{D\lambda^2 L}} \exp\left(-j\frac{\pi c T^2}{D\lambda^2 L}k^2\right) - \left\lfloor \frac{N}{2} \right\rfloor \leq k \leq \left\lfloor \frac{N}{2} \right\rfloor, \tag{4.12}$$

with $N = 2\lfloor 2\pi|M| \rfloor + 1$. A lower bound for the length of CD filters can be obtained by calculating the group delay difference induced by CD over the signal bandwidth B and distance L. In samples,

$$T_{\mathrm{CD}} = 2\pi \left|\beta_2\right| LB/T. \tag{4.13}$$

The direct sampling method does not approximate well the impulse response of chromatic dispersion when the accumulated dispersion is low (less than 320 km of SSMF). This is due to the aliasing resulting from an imprecise approximation of the CD transfer function time window[141]. More advanced CD filter design methods based on least-squares optimisation have been proposed [28, 121]. This approach minimises the energy of the complex error, defined in the frequency domain between the ideal CD transfer function (Eq. (4.6)) and the DFT of the desired FIR filter. In this thesis, we have used the LSCO method [121], proposed for the compensation of band-limited signals. Aside from making the in-band gain response as close as possible to ideal dispersion compensation, this technique constrains the out-of-band gain of the filter that results from reducing the number of filter taps. The method involves two stages: First, the filter taps are calculated. Second, find a threshold

that reduces the in-band error considering the coefficient quantisation errors induced by the out-of-band gain. The optimal FIR filter coefficients are given by

$$\tilde{h} = \boldsymbol{Q}^{-1}\boldsymbol{v}, \tag{4.14}$$

where $\boldsymbol{Q}$ is an $M \times N$ matrix with elements given by

$$Q_{m,n} = \begin{cases} \frac{2\pi(\lambda+1)+(\lambda+1)\Omega_1-(\lambda+1)\Omega_2}{2\pi+\Omega_1-\Omega_2} & \text{if } m = n \\ \frac{\lambda}{j(m-n)(2\pi+\Omega_1-\Omega_2)}\left[e^{j(m-n)\Omega_1} - e^{j(m-n)\Omega_2}\right] & \\ +\frac{1}{j(m-n)(\Omega_1-\Omega_2)}\left[e^{j(m-n)\Omega_1} - e^{j(m-n)\Omega_2}\right] & \text{if } m \neq n \end{cases} \tag{4.15}$$

and $\boldsymbol{v}$ is a vector given by

$$v_m = \frac{e^{-j\left(\frac{m^2}{4M}+\frac{3\pi}{4}\right)}}{2(\Omega_2-\Omega_1)}\sqrt{\frac{\pi}{M}}\left\{\text{erfz}\left[\frac{e^{-j\frac{\pi}{4}}(2M\Omega_1+m)}{2\sqrt{M}}\right] - \text{erfz}\left[\frac{e^{-j\frac{\pi}{4}}(2M\Omega_2+m)}{2\sqrt{M}}\right]\right\}. \tag{4.16}$$

Here, Ω_1 and Ω_2 are the boundary frequencies of the signal band, erfz is the error function of complex arguments and λ is a Lagrangian parameter. Figure 4.6 shows the response of LSCO filters of varying lengths, designed for a 32 Gbaud signal at 2 SpS, transmitted over a fibre of length $L = 100\,\text{km}$ with a dispersion parameter $D = 17\,\text{ps}/(\text{nm}\cdot\text{km})$. We observe that the length of the filter significantly influences its bandwidth and the phase response. The 59-tap filter shows a mostly flat gain profile, while shorter filters show greater oscillatory behaviour within the band. Gain peaks outside the band are observed, with oscillations that also intensify with shorter filter lengths. Equation (4.13) gives $T_{\text{CD}} \approx 31$, almost half the length required by the LSCO filter to show a flat gain profile. While this method provides filters which response is far from ideal when their length approaches the memory length given by Eq. (4.13), it may be acceptable for initialisation of short CD filters in nonlinear equalisers. While employing filters with gain ripples in a sequence would result in coherent build up of errors, this can be mitigated through gradient-based optimisation, which optimises the joint response of the filters [58].

Fully time-domain implementations have also been proposed for eliminating repeated Fourier transformations between linear and nonlinear stages in DBP-based multichannel

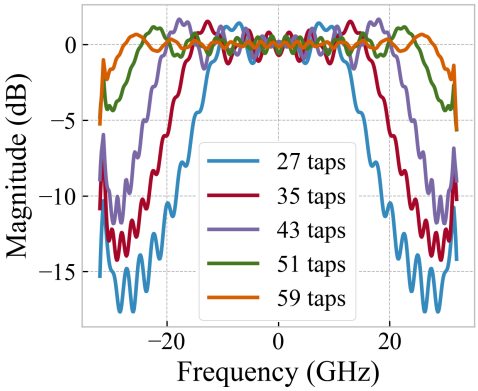


Figure 4.6: Magnitude response of LSCO filters of various lengths for a 32 Gbaud signal at 2 SpS and an SSMF fibre length of 100 km.

equalisation [67, 124]. This requires fully implementing the linear steps (i.e. Eq. 4.9) in the time domain. However, the walk-off delays $D_i(\omega)$ are generally fractional, and implementing fractional delays in the time domain is challenging. This issue can be addressed by treating delay as a resampling process that maintains a constant sampling rate. This approach has resulted in a variety of design methods for fractional delay filters suitable for various precision requirements [81]. In the design of fractional delay filters, our objective is to approximate the frequency response of $D_i(\omega)$, which has unity magnitude and constant group delay

$$|D_i(\omega)| = \left|e^{-j\beta_2 L\omega\Delta\omega_i}\right| = 1 \quad \text{for all } \omega \tag{4.17}$$

and

$$-\frac{d}{d\omega}\arg[H_d(\omega)] = \beta_2 L\Delta\omega_i = \tau. \tag{4.18}$$

For a band-limited baseband signal, the ideal solution can be obtained using the discrete-time inverse Fourier transform [103]

$$h(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} D_i(\omega)e^{j\omega n}d\omega = \text{sinc}(n-\tau) \quad \text{for all } n. \tag{4.19}$$

This solution is infinite in duration and non-causal and, therefore, cannot be implemented in our schemes. An approximation can be obtained using an N-th order FIR filter (of length $L = N + 1$), whose coefficients can be determined using a least squared error design approach. The L_2 optimal FIR of order N is derived by truncating the ideal solution to L

terms. The least squares solution can be enhanced to reduce the Gibbs phenomenon, which introduces ripples in the magnitude response, by applying a bell-shaped window function. This approach mitigates peak magnitude errors by smoothly tapering the signal, thereby minimizing discontinuities at the boundaries. Commonly used window functions include the Hamming, Hanning, and Blackman windows, each designed to suppress spectral leakage and reduce oscillations near discontinuities. In that case the overall impulse response becomes:

$$h(n) = \begin{cases} W(n-\tau)\text{sinc}(n-\tau) & \text{for } 0 \leq n \leq N \\ 0 & \text{otherwise.} \end{cases} \tag{4.20}$$

In our work we made use of a Hamming window, given by

$$W(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N}\right). \tag{4.21}$$

The window's shape and its frequency response are depicted in Fig. 4.7. We selected the windowed sinc method for its straightforward implementation, though it offers limited control over the magnitude response through parameter adjustments. In contrast, authors in [67] employed Lagrange interpolators, which provide a maximally flat delay response, making them suitable for applications requiring precise magnitude control. However, our chosen method is not recommended for filter lengths shorter than ten taps when stringent magnitude error control is necessary [81]. In our case, we assume that any magnitude deviations are not critical, as they can be compensated for by accompanying trainable filters.

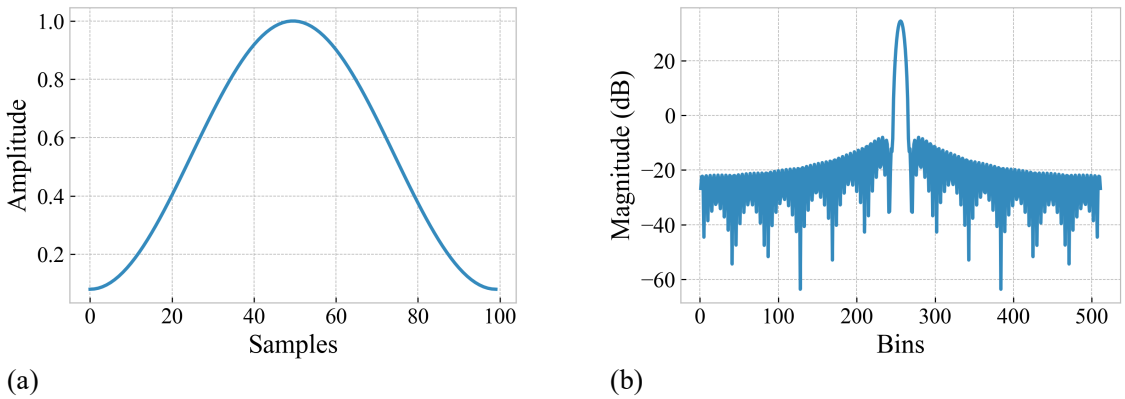


Figure 4.7: (a) Shape and (b) frequency response of the Hamming window used in the design of fractional delay filters.

In our algorithms, we may address substantial delays. For instance, consider the edge channel of a group of 9 subchannels with 40 GHz spacing, where $\Delta\omega_4 = 2\pi \cdot (160\,\text{GHz})$. Assuming $\beta_2 = -2.1683 \times 10^{-26}\,\text{ps}^2/\text{km}$, $L_\text{s} = 100\,\text{km}$ and $T = 15.625\,\text{ps}$, the group delay calculated using Eq. (4.18) is $\tau = -2.1798\,\text{ns}$, equating to approximately -139.50 samples. To manage such delays, we implement fractional delays using a combination of unit delays and a short fractional delay filter. This design offers flexibility in applying fractional delay filtering within schemes that involve sequential linear steps. For example, the approach in [67] addresses only integer delays in the linear steps of the LDBP algorithm, deferring fractional delay compensation to a separate layer at the end. In contrast, our algorithms handle fractional delays at each linear step. By integrating fractional delay filtering throughout the process, we achieve more precise delay compensation, enhancing the overall performance of our system.

The choice between time-domain and frequency-domain linear filtering depends on implementation requirements such as computational cost and latency. Frequency-domain filtering leverages the highly efficient FFT algorithm but incurs the drawback of discarding samples due to overlap-based block processing methods. On the other hand, time-domain filtering avoids the overhead of transforming signals to and from the frequency domain and eliminates the need for overlap-based processing. However, it can become computationally expensive when dealing with filters that have long impulse responses. A comparison of these methods for CD equalization is provided in [141], demonstrating that frequency-domain filtering is more efficient for handling large amounts of accumulated dispersion. However, a subsequent study [38] showed that time-domain chromatic dispersion compensation can be a power-efficient alternative for systems with moderate amounts of accumulated dispersion (up to approximately 150 km of SSMF). This scenario is common in nonlinearity equalization schemes, where the algorithm's linear steps compensate for the dispersion of shorter fiber sections. Examples include time-domain DBP [39] and time-domain Volterra series equalizers [56]. Given that efficient linear filtering implementations are feasible using both approaches, we have explored both time-domain and frequency-domain techniques for implementing the schemes presented in this thesis.

Matched Filtering

The spectral efficiency (SE) of modulated signal transmission is directly influenced by the pulse shape used. For AWGN channels, the Nyquist criterion specifies the conditions for pulse shapes that maximise spectral efficiency and minimise intersymbol interference. The sinc function is the optimal shape that satisfies the Nyquist criterion.

Although an analogous optimal pulse shape has not been derived for the nonlinear fibre channel, Nyquist pulse shaping remains an essential component in optical fibre networks as it effectively improves the SNR. Since the sinc impulse response is infinite in duration and impractical to implement, matched filtering is commonly achieved using root-raised cosine (RRC) filters characterised by a roll-off factor β. These filters are applied at both the transmitter and receiver, with an impulse response given by

$$f(t) = \begin{cases} \frac{1}{T}\left(1+\beta\left(\frac{4}{\pi}-1\right)\right) & \text{for } t = 0 \\ \frac{\beta}{T\sqrt{2}}\left((1+\frac{2}{\pi})\sin(\frac{\pi}{4\beta}) + (1-\frac{2}{\pi})\cos(\frac{\pi}{4\beta})\right) & \text{for } t = \pm\frac{T}{4\beta} \\ \frac{1}{T}\frac{\sin(\pi\frac{t}{T}(1-\beta))+4\beta\frac{t}{T}\cos(\pi\frac{t}{T}(1+\beta))}{\pi\frac{t}{T}\left(1-(4\beta\frac{t}{T})^2\right)} & \text{otherwise,} \end{cases} \tag{4.22}$$

where T is the sample period and t is the tap index.

4.2.2 Data Preprocessing for Model Training

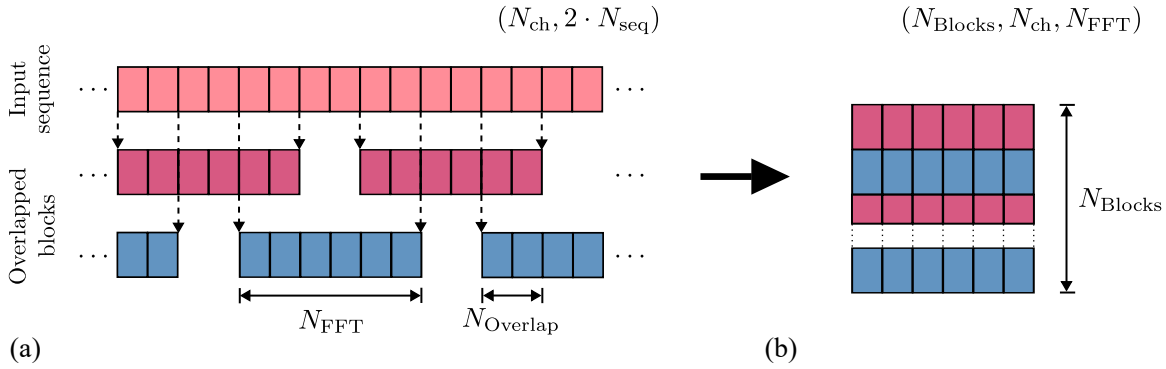


Figure 4.8: Depiction of (a) overlap and save multichannel pre-processing and (b) the dimensions of the resulting array.

In this section, we outline the preprocessing steps applied before model training and how they transform the data dimensions. While the focus is on multichannel processing, these steps are equally applicable to single-channel scenarios. Initially, the received data,

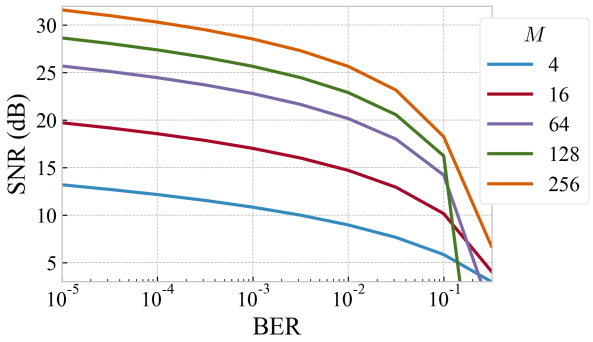


Figure 4.9: Effective SNR against the BER for different QAM orders.

sampled at 2 samples per symbol has dimensions $(N_{\text{ch,tx}}, N_{\text{samp}})$, where $N_{\text{ch,tx}}$ is the number of transmitted channels and N_{samp} is the length of the oversampled received sequence. First, the channels which are not be processed are discarded, resulting in an array of shape $(N_{\text{ch}}, N_{\text{samp}})$. Then, overlap-and save preprocessing is done for each channel, transforming the data array dimensions as shown in Fig. 4.8. Finally, a `tf.data.Dataset` object is created by pairing training data with the corresponding reference symbols. The training dataset is shuffled and divided into mini-batches of size N_{batch}.

Custom training loops were implemented based on the guidelines in [132], allowing precise control over the data flow and data formatting inside the training and evaluation steps. This was required due to the model's integration within an overlap-based multichannel DSP pipeline. As required by the overlap and save algorithm, samples were discarded before error calculation. In the validation step, errors were computed both as an average across all processed channels and for each channel to facilitate performance monitoring. To monitor training progress, we have monitored an SNR calculated from the mean squared error as $\text{SNR}_{\text{MSE}} = -10\log(\text{MSE})$. We use this metric in the training and validation steps performed at each epoch to provide a coarse indication of training progress. It has the advantage that can be computed quickly, which is highly desirable during the training stage. Conversely, during the test phase, we performed hard-decision symbol demodulation and computed the BER.

The performance results of this thesis are presented in terms of an effective SNR obtained from the BER (Eq. (2)). Figure 4.9 shows the relationship between the calculated effective SNR and the BER for different square QAM orders.

4.2.3 MIMO Processing for Equalisation

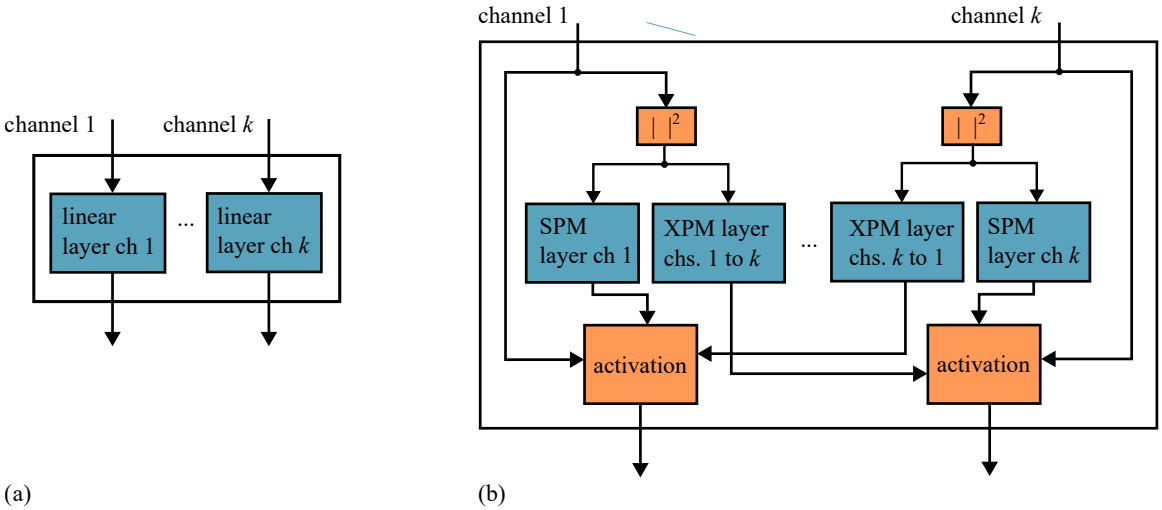


Figure 4.10: Block diagram of the WDM (a) linear and (b) nonlinear layers, showing channel 1 and channel k. In (a), the linear layers for each channel may be defined in the frequency or time domain. In (b), the SPM and XPM layers are implemented in the time domain.

The MIMO equalisers developed in this thesis are implemented using custom layers, defined by subclassing the `tf.keras.layers.Layer` class. In each custom layer, parameters are defined in the `__init__()` method, while forward passes are performed in the `call()` method. MIMO implementations employ a hierarchical architecture that includes single-channel layers and multichannel layers, offering flexibility in handling varying numbers of processing inputs: Multichannel operation can be reduced to single-channel operation through parameter configuration. Since IVSTF-based equalisers consist of linear and nonlinear stages, linear and nonlinear multichannel layers are defined as building blocks that can be concatenated to form various models. Employing single channel layers within multichannel layer supports weight tracking and retrieval. Multichannel linear layers are designed so that they may use time or frequency-domain single-channel layers, whereas multichannel nonlinear layers are to use time-domain single-channel layers exclusively. Diagrams illustrating the `call()` methods of the `wdm_linear_layer` and `wdm_nonlinear_layer` classes are shown in Figure 4.10, highlighting the internal differences between multichannel linear and nonlinear layers. Linear layers feature no interconnections between channel paths, while nonlinear layers require such interconnections to account for XPM contributions. By enclosing interconnections and complexity within multichannel layers, they can be employed as single-channel layers, permitting their reusability for constructing equalisation schemes

with varying structures. For example, although these layers were designed for Volterra-based equalisers, they can be used for implementing MIMO DBP. The structure of the `MIMO_equaliser`, `wdm_linear_layer` and `wdm_nonlinear_layer` classes is detailed below.

The pseudocode for the equalisation algorithms referenced in the `call()` method of the `MIMO_equaliser` class, and which are introduced in 6, is presented in Appendix C. Additionally, the Python implementation of single-channel layers is provided in Appendix D.

```
Class: wdm_linear_layer
Method: __init__()
  for channel in channels:
    Initialise linear layer with CD filters
Method: call()
  for channel in channels:
    Apply corresponding linear layer
```

```
Class: wdm_nonlinear_layer
Method: __init__()
  for channel in channels:
    Initialise SPM layer
  for each channel pair:
    Initialise XPM layer
Method: call()
  for channel in channels:
    Compute power signal
    Apply SPM layer to power signal
    Apply XPM layers to each channel's
    power signal
    Compute total nonlinear phase shift
```

Figure 4.11: Structure of the `wdm_linear_layer` (left) and `wdm_nonlinear_layer` (right) classes used to implement MIMO equalizers. Linear layers process each channel independently, while nonlinear layers employ inter-channel connections to account for cross-phase modulation effects.

```
Class: MIMO_equaliser
Method: __init__()
  for each computation step:
    Calculate CD, SPM, and XPM FIR coefficients
    Create wdm_linear_layer layer(s)
    Create wdm_nonlinear_layer layer
Method: call()
  Apply FFT to signal (if applicable)
  Apply the equalisation algorithm
  Apply iFFT to signal (if applicable)
```

Figure 4.12: Structure of the `MIMO_equaliser` class, which defines a multistep algorithm composed of linear and nonlinear layers. Each step computes the required FIR filter coefficients and instantiates the required layers. The `call()` method applies the equalisation algorithm in the time or frequency domain.

4.3 Summary

This chapter outlined the simulation and signal processing methods used for numerically assessing equalisation in single channel and multichannel scenarios. This was followed by a discussion of the various filtering techniques employed in model-driven equalisation, including time and frequency domain filtering techniques. Lastly, we introduced the framework developed for MIMO equalisation that supports the training of model-based MIMO schemes. The framework follows a layered approach for implementing MIMO equalisers, which is scalable and may be employed to investigate varying equalisation architectures.

Chapter 5

Learned Volterra Equaliser for Single Channel Transmission

The work presented in this chapter has been adapted from the following publication:
[12] N. Castro and S. Sygletos. A novel learned Volterra-based scheme for time-domain nonlinear equalization. In *Conference on Lasers and Electro-Optics*, page SF3M.1, San Jose, California, 2022. Optica Publishing Group. ISBN 978-1-957171-05-0.

5.1 Introduction

The integration of ML into optical communications has prompted a re-examination of conventional equalisation methods to enhance their performance and reduce complexity. A key example of this is the LDBP scheme, which transforms the time-domain DBP algorithm into an optimizable computation graph. This approach leverages the adaptability of trainable time-domain filters, enabling effective joint optimization that improves performance while reducing complexity, particularly by allowing for shorter filter lengths within the system. The IVSTF equalizer [88], a model-based alternative to DBP, is particularly attractive for hardware implementation due to its ability to compute, in parallel, the nonlinear distortion induced by each fiber span. This parallel computation allows for potential low-latency and high-throughput operation in low-complexity hardware [111]. However, several factors limit the IVSTF's performance. Since it is based on a truncated Volterra series, its ability to accurately model nonlinear effects is restricted. This manifests in the highly nonlinear power regime, where the performance of the algorithm is notably poor. Furthermore, while

in SSF-based algorithms increasing the number of steps leads to enhanced accuracy, employing sub-span steps in IVSTF leads to very limited accuracy improvements [27]. The simplifications necessary for straightforward implementation further limit performance. The structure treats nonlinearity and dispersion separately within the nonlinear contribution of each fibre segment, which enables sequential filtering stages. Consequently, regardless of the number of step used, the model fails to account for the interaction between nonlinearity and dispersion within each fiber span, leading to suboptimal performance.

In this chapter we investigate ML optimisation to address these limitations. We explore how to achieve effective joint optimisation of the filtering steps of the model. We propose a novel time domain scheme, the **L-simIVSTF**, which addresses the computational redundancy of the IVSTF by employing efficient FIR-based filtering. Our study demonstrates the effective gradient-based optimisation of the IVSTF-based architecture, enabling it to provide the same performance improvement as LDBP.

5.2 An IVSTF-based Machine Learning Model

We consider the development of a trainable model based on the IVSTF as an alternative to both LDBP and generic neural networks. The objective is to obtain a model-driven ML scheme whose architecture aligns with the IVSTF, with hyperparameters derived directly from the physical properties of the link. Suitable parameters in the model are to be optimised using supervised learning, an approach shown to be applicable to function classes beyond neural networks [58]. We first examine the LDBP model. The development of LDBP involved parameterising the SSF method to transform it into a model that can be trained as a NN. Conventional SSF-based DBP implementations employ identical filters $A_{\Delta z}$ at each linear step [72, 117]. A step i of the asymmetric SSF method is expressed as

$$\boldsymbol{u}_i = \boldsymbol{\sigma}_{\Delta z}(A_{\Delta z}\boldsymbol{u}_{i-1}) \quad \text{for } i = 1, 2, \ldots, M, \tag{5.1}$$

where M is the total number of steps. In contrast, LDBP allows the filters at each step to be distinct, similar to the weights matrices in a NN:

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{\sigma}_\ell(A^{(\ell)} \ldots \boldsymbol{\sigma}_1(A^{(1)}\boldsymbol{x})), \tag{5.2}$$

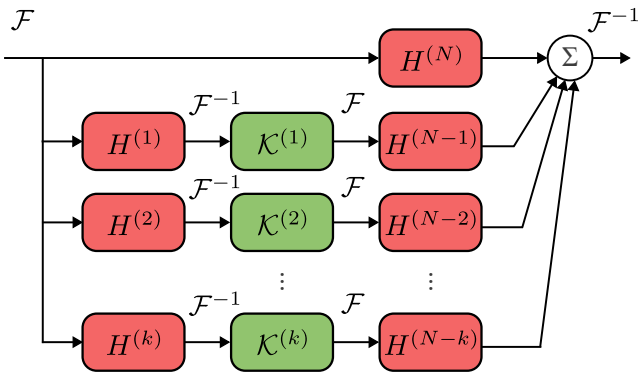


Figure 5.1: IVSTF model, where $H^{(k)}$ is a transfer function corresponding to the dispersion of k spans of fibre, and $\mathcal{K}^{(k)}$ is a nonlinear operator estimating the nonlinear phase shift of the k-th fibre segment.

where $\ell = M$ is the number of steps of the model. To limit the number of parameters, the matrices $A^{(\ell)}$ are restricted to symmetric FIR filters. The set of all trainable parameters of LDBP is represented as $\{\boldsymbol{A}^{(1)} \ldots, \boldsymbol{A}^{(\ell)}\}$. We adopt a similar approach to parameterise the IVSTF. While the IVSTF does not closely resemble deep NNs, it is similarly comprised by sequences of linear and nonlinear steps, where trainable parameters may be introduced. The IVSTF or first-order RP model can be expressed as a sum of two terms [27]:

$$U(L,t) \approx U_{\text{ln}}(L,t) + U_{\text{nl}}(L,t). \tag{5.3}$$

The term U_{ln} is a linear branch where the accumulated chromatic dispersion of the link is addressed:

$$U_{\text{ln}}(L,t) = \mathcal{D}^{(N)}\left[U(0,t)\right]. \tag{5.4}$$

Here, $\mathcal{D}^{(s)}$ is a linear operator addressing the dispersion of s fibre spans and it is given by

$$\mathcal{D}^{(s)}[\cdot] = \mathcal{F}^{-1}\left[H^{(s)}(\omega)\mathcal{F}\left[\cdot\right]\right]. \tag{5.5}$$

Here ω is the angular frequency, and $\mathcal{F}$ and $\mathcal{F}^{-1}$ represent the DFT and inverse DFT, respectively. Considering a 1 StpS implementation, $H^{(s)}(\omega)$ is a CD transfer function covering

s steps of length L_{sp},

$$H^{(s)}(\omega) = \exp(j\beta_2 s L_{\mathrm{sp}}\omega^2/2). \tag{5.6}$$

The term U_{nl} is the summation of nonlinear "branches" corresponding to the contributions from each segment k of fibre:

$$U_{\mathrm{nl}}(L,t) = \sum_{k=0}^{N-1} \mathcal{D}^{(N-k)} \left[\mathcal{K}^{(k)} \left[\mathcal{D}^{(k)} \left[U(0,t)\right]\right]\right], \tag{5.7}$$

where the operator $\mathcal{K}$ is defined as

$$\mathcal{K}^{(k)}[U(t)] = j\gamma L_{\mathrm{eff}}|U(t)|^2 U(t). \tag{5.8}$$

Each nonlinear branch in Eq. (5.7) consists of three filtering operations: the first is a linear step compensating for the dispersion from the beginning of the link to the k^{th} fibre segment, $k = 1...N$. The second is a nonlinear transformation mitigating the nonlinear phase shift corresponding to the k-th segment. Finally, a linear step addresses the dispersion from the end of the k-th segment to the end of the link. We note that the above expressions are a simplified version of the RP model which does not account for the lump losses in the backpropagation link. We refer the interested reader to Appendix A for a more accurate and detailed derivation of this implementation. We first inspect the available filters in the structure. The innermost linear operator in Eq. (5.7) employs a group of filters $\{H^{(0)}(\omega), H^{(1)}(\omega), \ldots, H^{(N-1)}(\omega)\}$, while the outermost linear operator uses $\{H^{(1)}(\omega), H^{(2)}(\omega), \ldots, H^{(N)}(\omega)\}$. Including the $H^{(N)}(\omega)$ filter required for U_{ln}, the structure utilises $2N$ filters in total, with each $H^{(k)}(\omega)$ filter appearing twice. One possible approach to parameterise the IVSTF involves assigning independent filters to each linear operator and defining optimisable parameters within each filter. However, for frequency domain transfer functions such as $H^{(s)}$, the suitable options for trainable parameters might be limited. In LDBP implementations using frequency domain linear steps, no trainable parameters are introduced [36], or the dispersion parameter has been designated as the sole trainable parameter [69]. The effectiveness of this approach is limited, resulting in models that provide subpar performance [36].

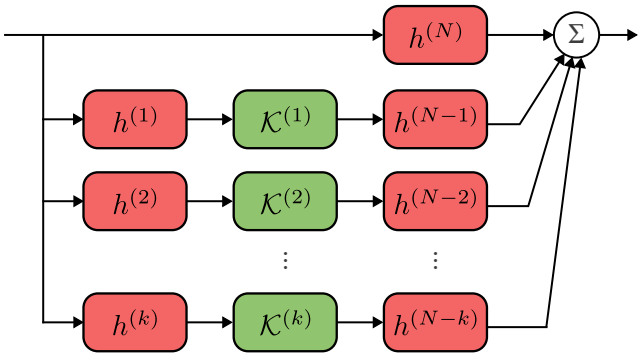


Figure 5.2: TD IVSTF model, where $h^{(k)}$ is a CD FIR filter addressing the dispersion of k spans of fibre. This model, a direct conversion of the IVSTF to the time domain, is highly inefficient.

A different parameterisation strategy was applied in [27], where the imprecision of the RP model was evaluated by examining the nonlinear phase rotation it induces. It was observed that when the nonlinearity of the optical signal is high, the nonlinear branches induce a gain increase that reduces the model's accuracy. The problem was then addressed by introducing trainable complex-valued vectors $\boldsymbol{C} = \{C_0, C_1, C_2, \ldots, C_N\}$ at the end of each branch, transforming the linear and nonlinear terms of Eqs. (5.4) and (5.7) into:

$$U_{\mathrm{nl}}(0,t) = \sum_{k=0}^{N-1} \mathcal{D}^{(N-k)} \left[\mathcal{K}^{(k)} \left[\mathcal{D}^{(k)} \left[U(L,t)\right]\right]\right] \times C_k. \tag{5.9}$$

These vectors enabled the adjustment of the gain and rotation of each branch. However, this only moderately improved equalisation performance. Yet another parameterisation approach is to perform linear steps in the time domain with FIR filters, enabling the FIR coefficients to be set as trainable parameters. Using this method to introduce trainable parameters in LDBP has shown superior performance gains from gradient-based optimisation compared to frequency-domain approaches [36]. Therefore, it may offer a promising direction for developing a learned IVSTF equaliser. Our next objective is to derive a time domain implementation of the IVSTF.

A time domain model equivalent to the IVSTF, which we refer to as the **TD IVSTF**, can be obtained by performing CD filtering in the time domain through convolutions with FIR filters. In this case, Eqs. (5.4) and (5.7) can be redefined as:

$$U_{\mathrm{ln}}(L,t) = h^{(N)} * U(0,t), \tag{5.10}$$

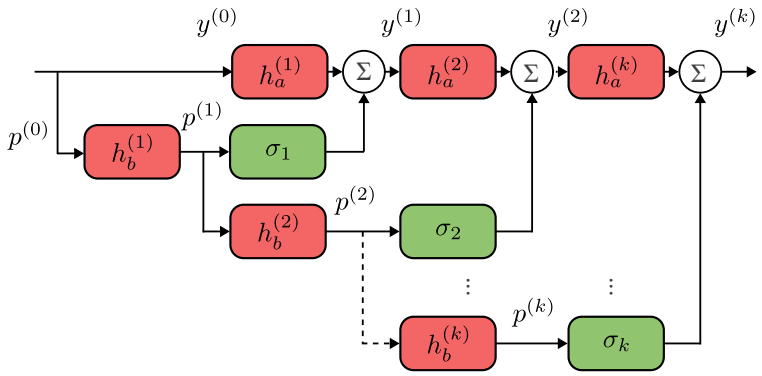


Figure 5.3: SimIVSTF structure that removes redundant filtering operations. The nonlinear activation at each step k is defined as $\sigma_k(x) \triangleq j\gamma L_{\text{eff}}|x|^2 x$, and $h_a^{(k)}$, $h_b^{(k)}$ are CD FIR filters associated with span k.

$$U_{\text{nl}}(L,t) = \sum_{k=0}^{N-1} h^{(N-k)} * \left(\mathcal{K}^{(k)}\left[h^{(k)} * U(0,t)\right]\right). \tag{5.11}$$

The corresponding scheme is shown in Fig. 5.2. The required filters $h^{(k)}$ address the dispersion of k fibre sections, and can be designed using the methods described in chapter 4. This implementation comes with significant drawbacks. The redundancy in CD equalisation across parallel arms results in inefficient scaling of computational complexity, as the FIR filters must address the dispersion of multiple fibre spans. Furthermore, there is a minimum length required for each filter to avoid performance penalty resulting from aliasing and account for pulse broadening. The length of these filters must be larger than the channel memory dictated by Eq. (4.13). For example, a filter $h^{(10)}$ addressing the dispersion of 10 fibre spans of 100 km requires at least 307 taps, assuming a bandwidth $B = (1+0.1)\cdot 32\,\text{GHz}$ and sample duration $T = 1/(2\cdot 32\,\text{GHz})$. Such filter lengths make convolutions prohibitely expensive. Moreover, the TD IVSTF implementation would require 2 of each $h^{(k)}$ filter, increasing overall cost and latency. Lastly, it is worth noting that optimising the TD IVSTF could be pursued by training the filter coefficients in a gradient-based optimisation process. While this path might lead to a highly adaptable learned scheme, the required convolutions would make its training highly inefficient. Therefore, further development of this scheme into a machine learning model has not been pursued.

The limitations of the TD IVSTF can be addressed with the simplified IVSTF scheme of Fig. 5.3. This approach employs interconnection of the nonlinear branches within an

IVSTF structure to enable the efficient reuse of short-length FIR filters to perform the CD compensation steps. Consequently, each nonlinear branch processes only the CD of its corresponding fibre section, eliminating redundant operations. Furthermore, since the FIR filters address the dispersion of a single step, all filters throughout the structure have the same low length requirement, enabling more efficient convolutions. The scheme can interpreted as the interconnection of two separate arrays of cascaded filters, denoted by $\boldsymbol{h}_a$ and $\boldsymbol{h}_b$. A step of the simIVSTF algorithm is recursively described as follows:

$$p^{(s)} = p^{(s-1)} * h_a^{(s)} \tag{5.12}$$

and

$$y^{(s)} = y^{(s-1)} * h_b^{(s)} + \sigma_k\left(p^{(s)}\right), \tag{5.13}$$

where the nonlinear activation for each step k is defined as $\sigma_k(x) \triangleq j\gamma L_{\text{eff}}|x|^2 x$. In these equations, $h_a^{(k)}$, $h_b^{(k)}$ represent the filters for the k-th step. The input field $A(0,t)$ is fed into both the a and b filtering arrays, with $p^{(0)} = y^{(0)} = A(0,t)$, where $A(z,t)$ is the propagating field. The equalised signal at the output of the equaliser is $A(L,t) = y^{(N)}$.

5.3 Simulation Setup and Results

Signal equalisation was characterised numerically by considering two transmission scenarios: single-channel and WDM transmission. In the single channel case, we considered the transmission of a 16-QAM RRC pulse-stream modulated at 32 GBaud along a 10×100 km fibre system amplified by EDFAs of 5 dB noise figure. In the WDM case, we consider 11 single-polarisation wavelength channels carrying 64-QAM symbols over a shorter 6×100 km link with 4.5 dB noise figure EDFAs. A shorter distance was chosen for the WDM case to ensure acceptable performance in the transmission of higher order symbols. The spacing between wavelength channels was 40 GHz. The symbol rate was the same for both scenarios. Single-mode fibre was used, with each span characterized by a dispersion parameter $D = 17\,\text{ps}/(\text{nm}\cdot\text{km})$, nonlinear factor $\gamma = 1.3\,(\text{W}\cdot\text{km})^{-1}$, and loss coefficient $\alpha = 0.2\,\text{dB/km}$. To represent the effects of signal distortion during propagation with sufficient accuracy, an up-sampling factor of 6 samples-per-symbol (SpS) was employed in the single-channel case, while 32 SpS was required in the WDM case to accurately represent

the interchannel interactions affecting the multiplexed signal.

At the receiver, after low-pass filtering the signal is down-sampled to 2 SpS before directing it to the nonlinear equaliser (NLE). Subsequently, matched filtering, downsampling to 1 SpS and genie-aided phase offset correction were applied, as shown in Fig. 4.3. The entire DSP chain was implemented in TensorFlow designating only the FIR filters taps of the NLE as trainable parameters.

The update of the CD FIR coefficients in the LsimIVSTF is based on the gradient of the MSE loss:

$$L_{\mathrm{MSE}} = \frac{1}{K}\sum_{c=1}^{K}|s_{\mathrm{out},n}^{(c)} - \hat{s}_{\mathrm{out},n}^{(c)}|^2. \tag{5.14}$$

We employed the Adam optimiser with a 0.001 learning rate. For the training we considered 15k gradient descent iterations conducted over a training set of 192×10^6 symbols. The final performance was characterised in terms of an effective SNR (Eq. (2)) on a test set comprising 64×10^6 symbols.

The CD FIR filters are restricted to be symmetric and of odd length. The symmetry in the filters was imposed during training by applying the same trainable coefficients in both sides of the filter. Pruning was applied on the trainable filters to reduce their size: at predefined epochs, a filter from each array was randomly selected, and its two outermost taps were discarded [57]. Target lengths were established at the start of the training routine, which were used to define a schedule specifying the required number of pruning steps. This schedule determined the training iterations at which pruning occurred, ensuring pruning steps were evenly distributed throughout the training process. At each epoch, each filter was pruned once. During each pruning step, the selected filter was pruned by applying a mask: The filter was multiplied by an FIR filter of identical length to the FIR CD filter, which zeros in the positions corresponding to the taps to be discarded.

In this study we compare our model with LDBP, the established model-driven approach. The LDBP scheme we employed is based on the asymmetric SSFM [58], where each computational step consists of a linear step followed by a nonlinear step. Although the symmetric SSF, involving two linear steps and one nonlinear step, offers better computational efficiency, the asymmetric method was selected to ensure that the model has the same amount of steps as the L-simIVSTF, enabling a direct comparison. The scheme underwent the same

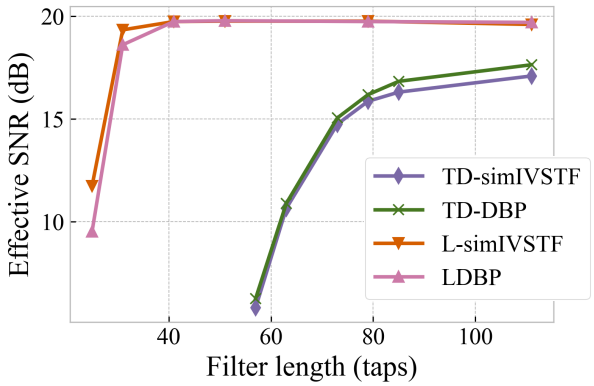


Figure 5.4: Comparison of equalisation performance in terms of effective SNR against filter length for the algorithms under consideration. ML optimisation enables a reduction in the length of the employed FIR filters, matching the minimum length required by LDBP.

initialisation, training and pruning procedures as the L-simIVSTF.

We first optimised the filter lengths of the L-simIVSTF using the previously described pruning technique. This optimisation process enabled us to assess how the model's training affects its minimum required length, and compare it with the filter length reductions achieved with LDBP. For comparison, we also optimised the filter length of the untrained time-domain equalisers, simIVSTF and DBP, by sweeping their filter lengths. Figure 5.4 shows the effective SNR performance at optimal launch power as a function of the filter length. For the untrained models, the performance follows a similar trend: it is poor for lengths close to the channel memory, and improves with the filter length until it eventually plateaus. The results indicate that more than 111 taps are required for both models to achieve optimal performance, yielding 17 dB for TD simIVSTF and 17.7 for TD DBP. For the learned schemes LsimIVSTF and LDBP, the performance also varies similarly but improves more rapidly compared to the unlearned schemes. Furthermore, the minimum required length for optimal performance is drastically reduced to 41 taps, where both models deliver a 19.8 dB performance. Notably, this minimum length is close to the minimum bound of 31 taps determined by the induced group delay.

We next examine the equalisation performance of L-simIVSTF in the single channel scenario. Figure 5.5 presents the effective SNR performance as a function of the channel launch power. All the equalisation schemes considered are implemented in the time domain with a single step per span, using the optimal filter lengths determined previously. TD IVSTF and TD DBP perform similarly, achieving a 0.6 dB improvement over CDE. The

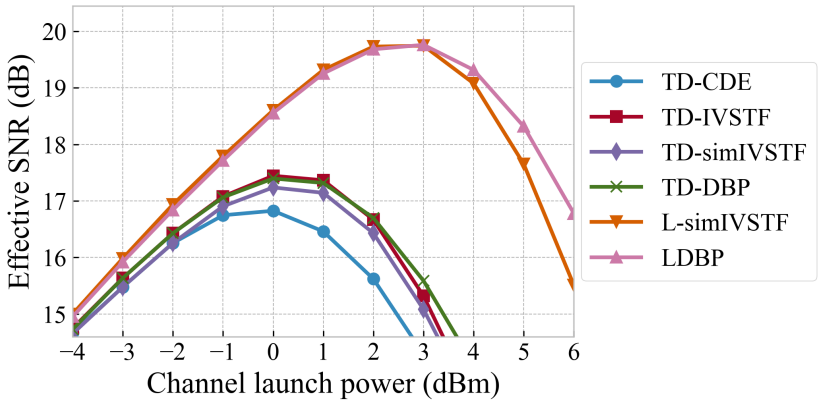


Figure 5.5: Comparison of equalisation performance in terms of effective SNR against channel launch power for the algorithms under consideration. L-simIVSTF shows an equivalent performance to LDBP.

non-optimised simIVSTF performs slightly worse due to truncation errors accumulating through the interconnected branches of the model. In contrast, the L-simIVSTF provides a 3 dB improvement over CDE, matching the performance of LDBP. Finally, we examine single channel equalisation in the WDM environment. Figure 5.6 shows effective SNR performance against launch power. As expected, the gains provided by L-simIVSTF are significantly reduced, offering only a 0.5 dB improvement over CDE.

5.4 Complexity Estimations

The computational cost of the proposed equaliser is evaluated in terms of real multiplications per transmitted symbol (RM/sym). This metric, commonly used for DSP algorithms, accounts only for multiplication operations while omitting additions [127]. This omission is justified, as the cost of multiplications is typically much higher than that of additions: multiplying two n-digit integers typically has a cost of $\mathcal{O}(n^2)$, while adding them has a cost of $\Theta(n)$ [42]. The complexity of TD-DBP is included as a reference.

Next, we calculate the complexity of simIVSTF. The cost of a linear step is determined by the convolution of the complex signal with a complex-valued filter of length S, requiring $4pS$ RM/sym, where p is the sampling rate. The nonlinear activations add $4p$ RM/sym due to the squared modules and multiplication by a complex constant. Therefore we have

$$C_{\text{simIVSTF}} = pN_s(8S + 4). \tag{5.15}$$

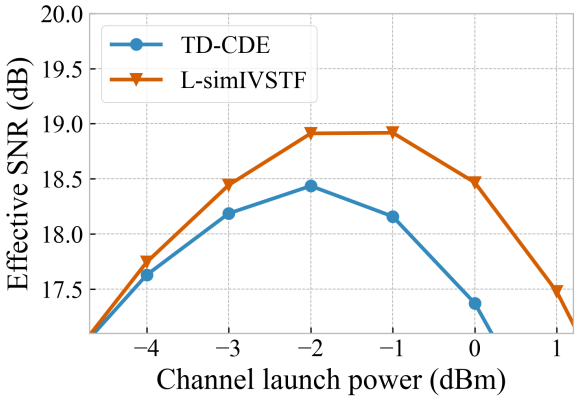


Figure 5.6: Performance of the L-simIVSTF model as a function of launch power in a single-polarisation, 6×100 km 11-channel WDM transmission scenario. The performance gain relative to CDE is limited to 0.5 dB.

The complexity of single-channel TD-DBP can be estimated by [124]

$$C_{\mathrm{TD-DBP}} = pN_s(4S + 4). \tag{5.16}$$

The complexity expressions for simIVSTF and TD-DBP are quite similar, primarily because the summations simIVSTF relies on have not been taken into account. As a result, the only difference in the expressions stems from simIVSTF using two cascades of N_s filters, while DBP employs only one. A more precise complexity estimation would need to consider the N_s summations in the structure. We observe that the complexity of simIVSTF is roughly twice the complexity of TD DBP.

We now compare the complexity of the proposed scheme to that of TD-CDE. The cost of TD-CDE using a complex-valued FIR filter of length S_{CDE} is $4pS_{\mathrm{CDE}}$ RM/sym. The minimum required length for a CD filter addressing the accumulated dispersion of the link can be estimated by considering the memory introduced by CD using Eq. (4.13). For a 10×100 km link, and assuming a bandwidth $B = (1 + 0.1) \cdot 32\,\mathrm{GHz}$ and sample duration $T = 1/(2 \cdot 32\,\mathrm{GHz})$, the minimum required length is 307 taps, resulting in a complexity of 2456 RM/sym. For the same link, the L-simIVSTF requires 41-tap filters to avoid performance penalties as shown in Fig. 5.4. With this length, Eq. (5.15) gives $C_{\mathrm{simIVSTF}} = 6640$ RM/sym, approximately 2.7 times the complexity of TD-CDE. Under the same assumptions, Eq. (5.16) gives a cost for LDBP of $C_{\mathrm{LDBP}} = 3360$ RM/sym, or

approximately 1.4 times the complexity of TD-CDE. This indicates that DBP achieves a more favourable performance-complexity trade-off relative to the proposed scheme.

5.5 Conclusions

We introduced a time-domain single-channel equaliser, L-simIVSTF, based on the simplification of the IVSTF architecture and enabled by machine learning. The performance of the scheme was assessed through numerical simulations, where the L-simIVSTF scheme demonstrated a 3 dB effective SNR improvement over CDE, matching the performance of LDBP. While the L-simIVSTF scheme proved more computationally intensive than its LDBP counterpart, it demonstrated the feasibility of gradient-based optimisation of Volterra-based equalisers.

However, in an 11-channel WDM transmission, the effective SNR improvement was limited to 0.5 dB. While this improvement is comparable to what can be obtained with other single-channel approaches in this scenario (single-channel LDBP showed less than 0.7 dB in a 5-channel WDM system [58]), it remains insufficient for practical scenarios. Therefore, the following chapter explores Volterra-based multichannel equalisation schemes, where greater performance gains are expected.

Chapter 6

Learned Volterra Equalisers for WDM Systems

The work presented in this chapter has been adapted from the following publications:

[13] N. Castro and S. Sygletos. Learned Volterra equalization for WDM systems. In *2023 Asia Communications and Photonics Conference/2023 International Photonics and Optoelectronics Meetings (ACP/POEM)*, pages 1–4, 2023.

[16] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Design of time-domain learned Volterra equalisers for WDM systems. In *2024 International Conference on Optical Network Design and Modeling (ONDM)*, pages 1–3, 2024. doi:10.23919/ONDM61578.2024.10582691.

[14] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Design aspects of frequency-domain learned MIMO Volterra equalisers. In *CLEO 2024*, page JTu2A.87. Optica Publishing Group, 2024.

[15] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Field-enhanced filtering in MIMO learned Volterra nonlinear equalisation of multi-wavelength systems. In *ECOC 2024; 50th European Conference on Optical Communication*, pages 902–905, 2024.

[17] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Learned Volterra models for nonlinearity equalization in wavelength-division multiplexed systems. *Opt. Express*, 33(8):16717–16737, Apr 2025. doi:10.1364/OE.554077.

6.1 Introduction

Over the past two decades, most studies on NLE have focused on developing schemes to address single-channel impairments [72, 108, 96, 98, 88, 122, 54, 56]. Although these approaches have achieved significant reductions in computational complexity, their effec-

tiveness in WDM scenarios has been limited. This limitation has shifted the focus of NLE design toward multi-channel operation. For multichannel equalisation, channel-by-channel processing based on a set of coupled[1] NLSEs with enhanced[2] nonlinear stages leads to MIMO NLE schemes as shown in [93, 94]. Compared to "full field" approaches [86], MIMO NLE schemes have a lower computational burden, made possible by allowing for larger step sizes and reduced sampling requirements. Nevertheless, exact knowledge of the transmission link parameters is still required for the *effective* operation of these schemes, and such a configuration process would be daunting for network operators.

Recently, ML has revolutionised the NLE field by introducing a variety of trainable algorithms. Model-driven approaches integrate the physical principles of signal propagation into the NLE operation, providing a framework that is easily understood and optimised. Early efforts in this field were based on perturbation models of the NLSE [126, 109], employing conventional machine learning techniques for training. Later, C. Häger et al. pointed out the functional similarity between the split-step Fourier method and DNNs in [58] and developed a learned version of a time-domain DBP algorithm that significantly enhanced the effectiveness of conventional DBP by incorporating DNN optimization capabilities. Following this, several other variants of the LDBP algorithm were created for both single [87] and multichannel operations [67, 69]. Despite the significant performance improvements and cost reductions, LDBP approaches depend on sequential computations of linear and nonlinear operations, which can result in excessive processing latency, as hardware parallelism is not easily leveraged to enhance efficiency and speed.

This chapter introduces learned multichannel equalisers based on the IVSTF model. It provides a unified overview of the developed architectures, including a theoretical derivation of the Volterra multichannel approach and a discussion of the physical assumptions underlying their design. The single-channel approach in Chapter 5 is extended to a MIMO configuration [13]. Additionally, time and frequency linear filtering techniques are leveraged to develop other architectures that maintain adaptability while reducing complexity [16, 14]. Gradient-based optimisation is employed to jointly optimise linear and nonlinear steps, enabling efficient multichannel equalisation. Computational efficiency is achieved

[1]Each equation in the set describes the propagation of a wavelength channel and accounts for the influence of adjacent channels [92].

[2]This term refers to the use of filtering strategies in nonlinear steps to better model the interaction between dispersion and nonlinearity [116].

through strategies such as FIR filtering of the power waveform for each channel. Design aspects crucial to improving performance and reducing training and inference costs, such as optimal initialisation and minimal required filter lengths, are examined. The chapter concludes with a performance and complexity analysis, addressing practical implementation aspects of each scheme. Our results and analysis may serve as guidelines for designing learned multichannel equalisers applicable in practical WDM systems.

6.2 Volterra Model

The volterra series (VS) provides a general mathematical framework to describe the non-linear behaviour of systems with memory. In the context of optical communications, they have been successfully applied to approximate the solution of the NLSE [104]. The primary difference from the SSF method is that VS separates the impact of fibre non-linearity and treats it in an additive manner, a critical feature for their low latency implementation when used as a backward propagation model in NLE applications. Although the VS framework for solving the NLSE has been developed in both the time [56] and frequency domains [3], only the frequency-domain approaches afford closed-form analytical expressions for the Volterra kernels.

Outlining the main steps involved in the derivation of our proposed IVSTF-based models for WDM signal transmission, we consider a multi-wavelength signal of $2K+1$ modulated channels of frequency spacing $\Delta\omega$ being transmitted along an optical fibre link. The signal propagation is described by the NLSE:

$$\frac{\partial A(z,t)}{\partial z} = -\frac{\alpha}{2}A(z,t) + j\frac{\beta_2}{2}\frac{\partial^2 A(z,t)}{\partial t^2} - j\gamma|A(z,t)|^2 A(z,t), \tag{6.1}$$

where $A(z=0,t) \triangleq \sum_{k=-K}^{K} A_k(t)e^{jk\Delta\omega t}$, $A_k(t)$ the complex field envelope of each wavelength channel, α is the propagation loss coefficient, β_2 is the group-velocity dispersion parameter, and γ is the non-linear parameter. At the end of the link the received signal is de-multiplexed and each sub-channel is detected coherently. We assume that the non-linear equalization is applied to a subset $2M+1$ ($M \leq K$) of the received sub-channels. The derivation of the proposed Volterra-based algorithm starts from the set of inverse coupled NLSE's given in Eq. (6.2), which describe the backward evolution of the baseband WDM

sub-channels being considered,

$$\frac{\partial U_m(z,t)}{\partial z} = (\hat{D}_m^{-1} + \hat{N}_m^{-1})U_m(z,t), \tag{6.2}$$

where $U_m(z,t)$ is the complex envelope of sub-channel m, $m = -M,\ldots,0,\ldots,M$, and $\hat{D}_m^{-1} = -j\frac{\beta_2}{2}\left(\frac{\partial^2}{\partial t^2} + 2jm\Delta\omega\frac{\partial}{\partial t} + (jm\Delta\omega)^2\right)$ and $\hat{N}_m^{-1} = \frac{\alpha}{2} + j\gamma\left(|U_m|^2 + 2\sum_{p\neq m}|U_p|^2\right)$ are the linear operators and nonlinear operators, respectively. Typically, Eq. (6.2) is solved using the SSF method by discretising the z-axis in a number of segments $[z_{n-1}, z_n]$ of step size $h_n = z_n - z_{n-1}$ with $n = 1...N$ and treating independently the linear and nonlinear propagation effects in each one of them. For the n^{th} step we can write:

$$U_m(z_n,t) = \exp\left[h_n(\hat{N}_m^{-1} + \hat{D}_m^{-1})\right]U_m(z_{n-1},t) \approx \exp(h_n\hat{N}_m^{-1})\underbrace{\exp(h_n\hat{D}_m^{-1})U_m(z_{n-1},t)}_{U_m^{\mathrm{d}}(z_n,t)}. \tag{6.3}$$

The order in which the linear and nonlinear operators appear in Eq. (6.3) may interchange. Applying the linear step first leads to the following analytic solution for the linear step in the frequency domain:

$$U_m^{\mathrm{d}}(z_n,t) = \mathcal{F}^{-1}\{\mathcal{F}[U_m(z_{n-1},t)]H_m(h_n,\omega)\}, \tag{6.4}$$

where $\mathcal{F}$ denotes the Fourier transform operation, and $H_m(h_n,\omega) = \exp(j\frac{\beta_2}{2}(\omega + m\Delta\omega)^2 h_n)$ is the multi-channel linear transfer function, in which the term $m\beta_2\Delta\omega$ is responsible for the walk-off effect among the different channels. The nonlinear step can then be analytically described in the time domain, thereby leading to the following form for Eq. (6.3):

$$U_m(z_n,t) = U_m^{\mathrm{d}}(z_n,t)\exp(\alpha h_n/2)\exp\left[j\phi_m^{\mathrm{nl}}(z_n,t)\right], \tag{6.5}$$

where $\phi_m^{\mathrm{nl}}(z_n,t) = \gamma h_n^{\mathrm{eff}}\left(|U_m^{\mathrm{d}}(z_n,t)|^2 + 2\sum_{p\neq m}|U_p^{\mathrm{d}}(z_n,t)|^2\right)$ is the total nonlinear phase shift attributed to the SPM and XPM effects, and $h_n^{\mathrm{eff}} = (\exp(\alpha h_n) - 1)/\alpha$ is an effective step size accounting for the influence of the effective gain on the signal envelope [96].

Equation (6.5) describes a MIMO equalization structure for fibre transmission links, where the linear and non-linear stages are alternated. While this approach can achieve high

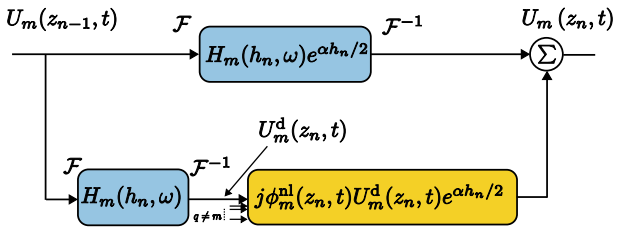


Figure 6.1: Diagram of the n-th step of a MIMO equalisation architecture based on the first-order polynomial expansion of the exponential function from the nonlinear step of the SSFM.

computational accuracy, it suffers from increased latency and computational complexity. Re-configuring the equalizer's architecture from a sequential to a completely parallel form instead can help overcome these challenges. This is the approach that we adopt here. Therefore, performing a 1$^{\mathrm{st}}$-order polynomial expansion of the nonlinear exponential function in Eq. (6.5) leads to [136]

$$U_m(z_n,t) = U_m^{\mathrm{d}}(z_n,t)\exp(\alpha h_n/2) + j\phi_m^{\mathrm{nl}}(z_n,t)U_m^{\mathrm{d}}(z_n,t)\exp(\alpha h_n/2), \tag{6.6}$$

where the first term addresses the impact of linear dispersion, while the second term accounts for the Kerr-induced nonlinear effects. Equation (6.6) provides a transfer function for the back-propagated signal over a single step, treating fibre nonlinearity as an additive perturbation, see Fig. 6.1. The concept can be extended to an entire transmission link by assuming that the nonlinear perturbation generated in each section does not contribute to the nonlinear processes of the subsequent sections. This is illustrated in Fig. 6.2 (a), for the case of three consecutive computational steps, part of a longer virtual back-propagation link. The blue line represents the signal propagation through the linear path, triggering the nonlinear process of each section. The corresponding perturbation terms that are generated, represented by red lines, follow the same linear path and add up only at the end of the link without affecting the nonlinearity generation processes of their subsequent sections. Using this key assumption, we can resolve Eq. (6.6) in a recursive manner and derive the following closed-form solution for the back-propagated signal field of the m channel after N sections:

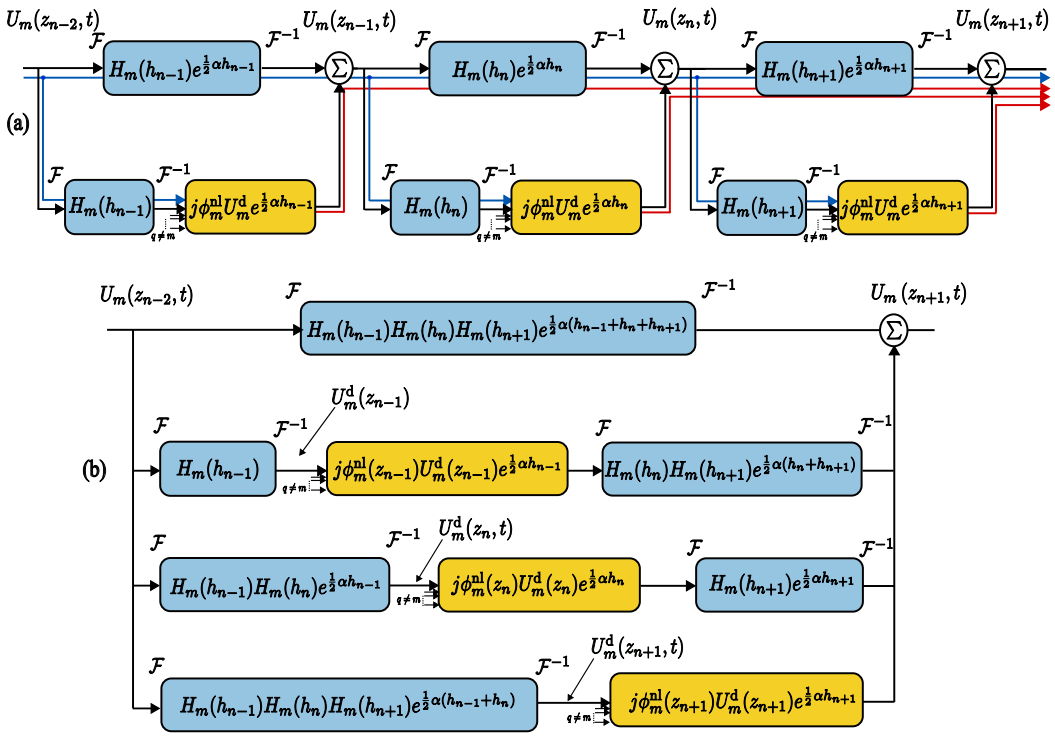


Figure 6.2: (a) Diagram for three consecutive computational steps of the MIMO equaliser, illustrating the propagation assumptions leading to a parallel architecture: The blue lines represent signal propagation through the linear path, which triggers the nonlinear processes of each fibre section. The red lines represent the propagation of the perturbation terms. (b) Equivalent parallel architecture.

$$U_m(z_N,t) = U_m^{\mathrm{ld}}(z_N,t)\exp(\alpha h_N/2) + j\sum_{n=1}^{N}\hat{\phi}_m^{\mathrm{nl}}(z_n,t)\exp(\alpha h_n/2)U_m^{\mathrm{ld}}(z_n,t) * \mathcal{F}^{-1}\left[\prod_{k=n+1}^{N} H_m(h_k,\omega)\exp\left(\alpha h_k/2\right)\right], \quad (6.7)$$

where $U_m^{\mathrm{ld}}(z_n,t)$ describes the linear part of the signal envelope at the end of the n^{th} section. This signal field has experienced the combined impact of dispersion and reverse loss of the previous $n-1$ sections, as well as the dispersion of the current step:

$$U_m^{\mathrm{ld}}(z_n,t) = U_m(z_0,t) * \mathcal{F}^{-1}\left[\prod_{k=1}^{n} H_m(h_k,\omega)\right]\exp\left(\frac{1}{2}\sum_{k=1}^{n-1}\alpha h_k\right), \quad (6.8)$$

and determines the SPM and XPM-induced non-linear phase shift in the n-th step, i.e. $\phi_m^{\mathrm{nl}}(z_n,t) \simeq \hat{\phi}_m^{\mathrm{nl}}(z_n,t) = \gamma h_n^{\mathrm{eff}}\left(|U_m^{\mathrm{ld}}(z_n,t)|^2 + 2\sum_{p\neq m}|U_p^{\mathrm{ld}}(z_n,t)|^2\right)$.

The final step in our mathematical derivation is to extend the single-span structure of Eq. (6.7), to an equaliser capable of dealing with multi-span systems. Inspecting Eq. (6.7) we can see that the first term represents the linear part of the BP process, responsible for equalising CD and loss effects, while the second term aggregates the nonlinear contributions of the BP steps, thereby counteracting the accumulated nonlinear effects from the forward direction. It is important to note that the nonlinear perturbation from each section n is convolved with the dispersion-induced time response of all the subsequent sections to the end of the link. For a multi-span system, the virtual BP link must include loss elements $L^{(q)}, q = 1, \ldots, Q$ between consecutive spans q to preserve power symmetry. Based on these considerations, the derivation of the overall transfer function of the multi-span equaliser is straightforward by applying the same assumption as in the single-span case, i.e., the nonlinear distortions generated in each span accumulate linearly over the subsequent spans without influencing the following span's nonlinear processes. Therefore, we can write:

$$\begin{aligned} U_m^{(Q)}(z_N,t) &= U_m^{\mathrm{ld}(Q)}(z_N,t)\cdot\exp\left(\frac{\alpha}{2}h_N^{(Q)}\right) \\ &\quad + j\sum_{q=1}^{Q}\sum_{n=1}^{N}\hat{\phi}_m^{\mathrm{nl}(q)}(z_n,t)\cdot\exp\left(\frac{\alpha}{2}h_n^{(q)}\right)\cdot U_m^{\mathrm{ld}(q)}(z_n,t) \\ &* \mathcal{F}^{-1}\left[\prod_{k=n+1}^{N} H_m^{(q)}(h_k,\omega)\exp\left(\frac{\alpha}{2}h_k^{(q)}\right)\cdot\prod_{s=q+1}^{Q}\left(\sqrt{L^{(s)}}\cdot\prod_{j=1}^{N}H_m^{(s)}(h_j,\omega)\exp\left(\frac{\alpha}{2}h_j^{(s)}\right)\right)\right], \end{aligned} \tag{6.9}$$

where the linearly propagating field $U_m^{\mathrm{ld}(q)}(z_n,t)$ in the q-th fiber span is given by

$$\begin{aligned} U_m^{\mathrm{ld}(q)}(z_n,t) &= U_m(0,t) * \mathcal{F}^{-1}\left[\prod_{s=1}^{q-1}\left(\sqrt{L^{(s)}}\cdot\prod_{k=1}^{N}H_m^{(s)}(h_k,\omega)\exp\left(\frac{\alpha}{2}h_k^{(s)}\right)\right)\right] \\ &\quad * \mathcal{F}^{-1}\left[\sqrt{L^{(q)}}\prod_{i=1}^{n-1}\left(H_m^{(q)}(h_i,\omega)\exp\left(\frac{\alpha}{2}h_i^{(q)}\right)\right)H_m^{(q)}(h_n,\omega)\right]. \end{aligned} \tag{6.10}$$

The calculation of the nonlinear phase shift of channel m at the n-th computational step described earlier accounts for the interaction between fibre loss and Kerr nonlinearity, but it neglects the influence of CD on the nonlinear dynamics of the step. Although this can

be effectively incorporated through an additional filtering operation on the power waveform of the channel, the exact formulation of the filter transfer function constitutes a challenge. Factorising the walk-off effect across the different MIMO channels led to an analytical expression for the frequency response of the XPM contribution [93]. For the SPM contribution, predefined window shapes were utilised in the "weighted DBP" scheme presented in [108], while brute force numerical optimisation of the filter transfer function was applied in the "enhanced DBP" scheme of [117]. In both cases, yet the filter response was identical across successive computational stages, causing the NLE architecture to accumulate truncation errors. Independent optimisation of the filter parameters at each stage through a gradient-based BP algorithm addressed the build-up of these errors, significantly improving the performance and allowing for larger computational step sizes [124, 58]. Building on these advancements, we enhanced the accuracy of our IVSTF-based models by applying signal power filtering in the nonlinear steps with filters trained as part of the whole optimisation process of the NLE algorithm. Therefore, the nonlinear phase shift for channel m at the q-th fibre span can be expressed as

$$\hat{\phi}_m^{\mathrm{nl}(q)}(z_n,t) = \gamma h_n^{\mathrm{eff}} \left(\sum_{c=-l}^{l} \mu_{m,c}^{(q,n)} |U_m^{\mathrm{ld}(q)}(z_n, t+cT_s)|^2 + 2 \sum_{p\neq m} \sum_{c=-k}^{k} \nu_{m,p,c}^{(q,n)} |U_p^{\mathrm{ld}(q)}(z_n, t+cT_s)|^2 \right), \quad (6.11)$$

where $\mu_{m,c}^{(q,n)}$ and $\nu_{m,p,c}^{(q,n)}$ are the coefficients of SPM and XPM FIR filters corresponding to the m-th's channel, and $S_{\mathrm{SPM}} = 2l+1$ and $S_{\mathrm{XPM}} = 2k+1$ are the respective filter lengths.

6.2.1 Architectural Variants

Equations (6.9)–(6.11) provide the mathematical foundation for the different IVSTF architectures presented in this paper, which are schematically illustrated in Figs. 6.3 and 6.4. The first variant is the L-IVSTF model shown in Fig. 6.3(b). Its key feature is that all linear field operations required to equalise CD effects are performed statically in the FD, while the nonlinear stages are trainable and executed in the TD. This design necessitates the use of at least one FFT/inverse FFT (IFFT) pair in each branch of the architecture, for which the

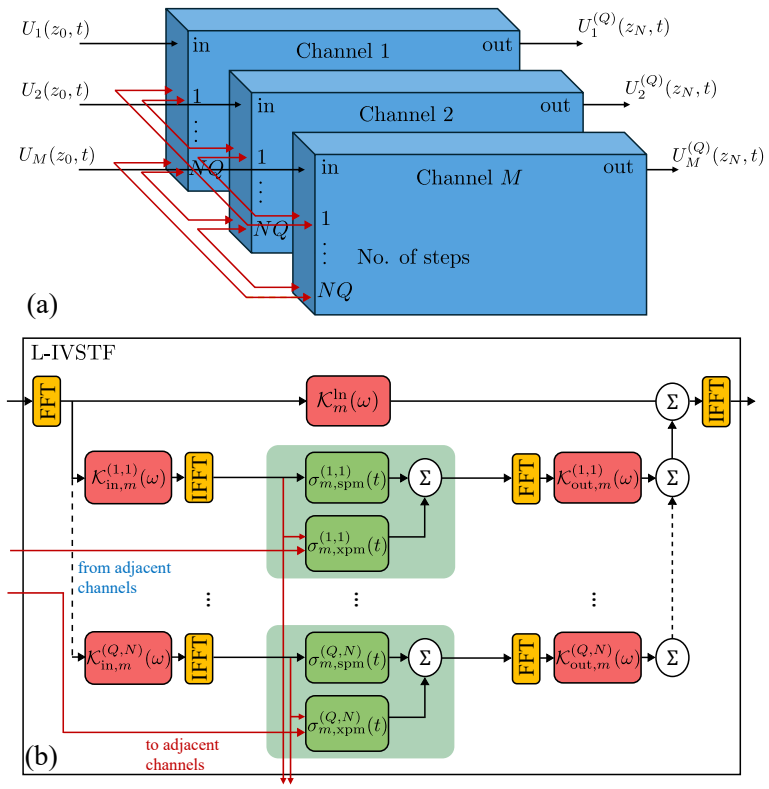


Figure 6.3: (a) Interconnection of channel processing units in the proposed MIMO schemes. (b) Processing units for channel n in the L-IVSTF model, depicting the 1st and k-th steps. Time-domain filtering is shown in shaded regions, with green areas representing the nonlinear steps and blue areas indicating linear steps.

block size N_{FFT} is a critical design parameter. Without loss of generality, we assume a fixed computational step h across the BP link and full equalisation of the signal power across consecutive spans. As a result, $H_m^{(s)}(h_n,\omega) = H_m(h,\omega)$, and the frequency transfer function of the upper branch becomes $\mathcal{K}_m^{\ln}(\omega) = H_m(h,\omega)^{QN}$. Accordingly, the input and output transfer functions of the parallel branch associated with the n-th step of the q-th span can be derived from Eqs. (6.9) and (6.11) as $\mathcal{K}_{in,m}^{(q,n)}(\omega) = e^{-(N-n+1)\frac{ah}{2}} H_m(h,\omega)^{(q-1)N+n}$ and $\mathcal{K}_{out,m}^{(q,n)}(\omega) = e^{(N-n)\frac{ah}{2}} H_m(h,\omega)^{(Q-q+1)N-n}$. Finally, using Eqs. (6.8) and (6.11), we can define the following TD transfer functions for the architecture:

$$\sigma_{m,\text{spm}}^{(q,n)}(t) = j\gamma h_n^{\text{eff}} U_m^{\text{ld}(q)}(z_n,t) \sum_{c=-l}^{l} \mu_{m,c}^{(q,n)} |U_m^{\text{ld}(q)}(z_n, t+cT_s)|^2, \tag{6.12}$$

$$\sigma_{m,\text{xpm}}^{(q,n)}(t) = 2j\gamma h_n^{\text{eff}} U_m^{\text{ld}(q)}(z_n,t) \sum_{p\neq m} \sum_{c=-k}^{k} \nu_{m,p,c}^{(q,n)} |U_p^{\text{ld}(q)}(z_n, t+cT_s)|^2. \tag{6.13}$$

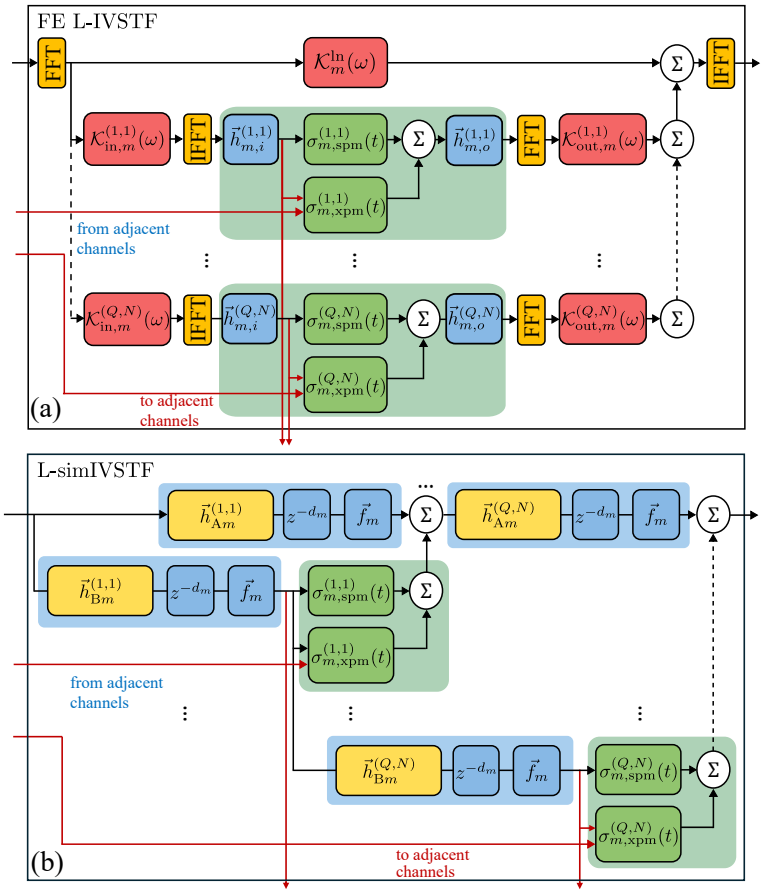


Figure 6.4: Processing units for channel n in the (a) FE L-IVSTF and (b) L-simIVSTF models, depicting the 1st and k-th steps. Time-domain filtering is shown in shaded regions, with green areas representing the nonlinear steps and blue areas indicating linear steps.

The L-IVSTF model in Fig. 6.3(b) relies on optimising only the FIR filters of the nonlinear stages. Thus, adaptability and performance are hindered by the absence of trainable linear steps. To overcome this limitation, we propose the FE L-IVSTF model shown in Fig. 6.4(a). In this scheme, short CD FIR filters, characterised by the coefficient vectors $\vec{h}_{m,i}^{(q,n)}$ and $\vec{h}_{m,o}^{(q,n)}$, are added at the input and output of each nonlinear step, enabling a joint fine-tuning of the linear response for all branches. Unavoidably, the FE learned inverse Volterra series transfer function (L-IVSTF) scheme introduces additional hyper-parameters, namely, the ratio between the dispersion managed statically and adaptively, and the required length of the FIR filters. Adequate values for these parameters can be found through simple parameter sweeps, as will be shown in Sec. 6.4.

The hybrid architecture of the L-IVSTF and FE L-IVSTF models relies on multiple

FFT/IFFT operations that add hardware complexity and computational cost. As discussed in Chapter 5, direct replacement of the FFT+filter+IFFT sections with time-domain FIR filters would not only require excessively long FIR filters but also cause inefficient scaling of the architecture's computational complexity since each branch would need to address the chromatic dispersion of the entire link. The simplified interconnection structure introduced in the previous chapter as the simIVSTF model [12] avoids the equalisation redundancy through efficient FIR filter re-use that allows each branch to deal only with the chromatic dispersion of its own computational step. An extension of this scheme to multi-channel operation, namely, a MIMO L-simIVSTF, is depicted in Fig. 6.4(b). The scheme comprises two independent FIR filter arrays, i.e., $\vec{h}_{\mathrm{A}m}^{(q,n)}$ in the linear and $\vec{h}_{\mathrm{B}m}^{(q,n)}$ in the nonlinear path of each computational step h. All filter coefficients are trainable, enabling the optimisation of the joint response of the filter arrays. However, these FIR filters do not address walk-off effects. For this purpose, a fractional delay filter $\vec{f}_m$ is employed in conjunction with a circular shifter z^{-d_m}. Under the assumption of a fixed computational step, both elements depend solely on the channel index m. The nonlinear stages of this architecture are given by Eq. (6.11).

Compensating exactly the group delay difference between the channels is critical for an accurate estimation of the inter-channel nonlinearity. With the L-IVSTF and FE L-IVSTF models, this can be done by adjusting the exponent in the frequency domain transfer function of the linear steps. Conversely, the task is more challenging with the L-simIVSTF model since the group delay may not coincide with the time grid of the digital sampling, thereby requiring further fractional adjustment. The group delay after propagation over a step, in terms of number of samples, is given by $\tau_m = 2\pi\Delta f_m \beta_2 h/T_s$, where T_s is the sampling period, and Δf_m the channel location relative to the central frequency. Depending on the channel position, this delay can be positive or negative and may have integer and fractional parts. For example, for a channel located at $\Delta f_1 = 40\,\mathrm{GHz}$ and assuming a group velocity dispersion parameter $\beta_2 = -2.1683 \times 10^{-26}\,\mathrm{ps}^2/\mathrm{km}$, fibre length $h = 100\,\mathrm{km}$ and a sample duration $T_s = 15.625\,\mathrm{ps}$, we obtain a delay $\tau_1 = -34.876$ samples. Its integer part $\lfloor \tau_m \rfloor$ can be corrected by a circular shifting operation, while the fractional part $\{\tau_m\}$ can be compensated with a short delay FIR filter. This filter is designed using the window method described in [78], where the filter taps are defined by the coefficient vector $\vec{f_m} \triangleq \vec{f}(\{\tau_m\}) = \left[\alpha W(t - \{\tau_m\})\mathrm{sinc}(\alpha(t - \{\tau_m\}))\right]_{t=0}^{T-1}$, with $t \in Z$ representing the

integer sample index [81]. For this implementation, we used a Hamming window defined as $W(t) = 0.54 - 0.46\cos(2\pi t/(T-1))$. The window length $T \in Z$ was chosen to minimise computational impact while being sufficiently large to avoid performance degradation. The bandwidth parameter α was set to 95%, ensuring reduced out-of-band gain. It is worth noting that addressing the fractional delay at each linear step may become impractical if the computational cost of the required filtering operations is excessively high. However, compensating for the integer delay using a series of unit delays while addressing only the sub-sample delay allows for relatively short filters, thereby reducing the computational cost of the fractional filtering operations. Furthermore, while designing very short fractional delay filters for processing high-baud-rate signals may introduce undesired in-band gain, these effects are expected to be mitigated by the accompanying trainable CD FIR filters.

6.3 Simulation Setup and Training Procedure

To assess the performance of our equalisers, we considered the transmission of 11 single-polarization wavelength channels over a link consisting of 6 × 100-km spans of standard single-mode fibre, with dispersion parameter $D = -2\pi c\beta_2/\lambda^2 = 17\,\mathrm{ps/(nm\cdot km)}$, nonlinear factor $\gamma = 1.3\,(\mathrm{W\cdot km})^{-1}$, and loss parameter $\alpha = 0.2\,\mathrm{dB/km}$. EDFAs with a noise figure of 4.5 dB compensated for the span losses. Each channel carried a stream of root-raised cosine pulses of 0.1 roll-off, modulated by 64 quadrature-amplitude modulation symbols at a rate of 32 Gbaud. The channel spacing was $\Delta\omega/(2\pi) = 40\,\mathrm{GHz}$. Data transmission was simulated using the SSF method in batches of $R = 2^{18}$ symbols, with an up-sampling factor of 32.

At the receiver, the channels of interest were de-multiplexed and down-sampled to 2 samples per symbol before being processed by the MIMO NLE. Following the NLE stage, each channel was match-filtered and further down-sampled to 1 sample per symbol. For the operation of the L-IVSTF and FE L-IVSTF models, we applied overlap-and-save block processing [101]. This was crucial for efficient handling of the linear filtering operations in the FD by segmenting the input signal into overlapping blocks, applying the FFT to each block, and then combining the results. The overlap length N_e and FFT size N_{FFT} were optimised to ensure performance and avoid penalties across all MIMO dimensions (see Sec. 6.4). The receiver's DSP blocks were implemented as a differentiable computation

graph in TensorFlow 2 to take advantage of the framework's extensive capabilities, such as automatic differentiation, GPU acceleration and flexible model tuning. During the training phase, the outputs of the MIMO NLE were linked to a single MSE function for computing the gradients of the model's trainable parameters,

$$L_{\mathrm{MSE}} = \frac{1}{MR}\sum_{m=1}^{M}\sum_{r=1}^{R}|s_{\mathrm{out},m}^{(r)} - \hat{s}_{\mathrm{out},m}^{(r)}|^2, \tag{6.14}$$

where $\hat{s}_{\mathrm{out},m}^{(r)}$ and $s_{\mathrm{out},m}^{(r)}$ are the reference and recovered symbols, respectively. During the testing phase, the recovered symbols from each channel were used to compute the bit error rate, which was then mapped to an effective signal-to-noise ratio (SNR) [119]. For a given launch power, the datasets included 2^{19} symbols for training and 2^{18} symbols each for validation and testing.

The Volterra models were trained using the Adam optimiser, where the initial training rate and batch size were tuned separately for the time-domain (TD; L-simIVSTF) and the frequency-domain (FD; L-IVSTF, FE L-IVSTF) models. The training of the FD models was generally more stable, thereby enabling a relatively large initial learning rate of 0.01, whilst training the TD model required an order of magnitude smaller training rate (0.001) to avoid divergence. The respective batch sizes were 25 and 40. MIMO schemes of varying sizes were trained separately for each launch power. Training was done over up to 1500 epochs, after which no further improvements were observed.

The trainable parameters included the real-valued coefficients $\mu_{m,c}^{(q,n)}$, $\nu_{m,p,c}^{(q,n)}$ of the SPM and XPM FIR filters, respectively, for all the models, and the complex-valued coefficients ($\vec{h}_{Am}^{(q,n)}$, $\vec{h}_{Bm}^{(q,n)}$) and ($\vec{h}_{m,i}^{(q,n)}$, $\vec{h}_{m,o}^{(q,n)}$) of the CD FIR filters for the L-simIVSTF and FE L-IVSTF models, respectively. The filters were carefully initialised to ensure the desired model's convergence. Specifically, the CD filters were initialised following the method in [121], while the initialisation of SPM and XPM filters was a subject of study. The model's hyper-parameters, including the lengths of all FIR filters, the amount of dispersion to be compensated by the CD FIR filters in the FE L-IVSTF model, and the FFT block length N_{FFT} and number of steps per span N in the FD models, were optimised to maximise performance and reduce complexity, as described in the following section.

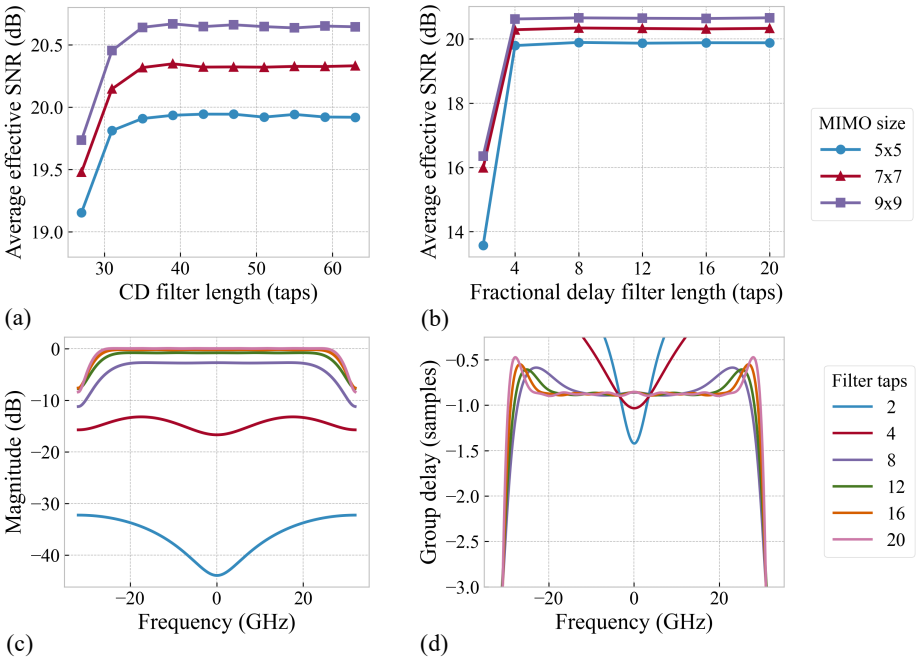


Figure 6.5: Filter length optimisation for the L-simIVSTF model. SNR performance of TD simIVSTF models with varying (a) CD filter lengths and (b) fractional delay filter lengths. Using 43-tap CD filters and 8-tap fractional delay filters ensures optimal performance. Response of fractional delay filters with varying lengths, approximating the fractional delay at a linear step (~ 0.876 samples), showing (c) magnitude response in dB and (d) delay in samples.

6.4 Results and Discussion

We conducted extensive numerical simulations to optimise the hyper-parameters and compare the performance of the different Volterra models. Hyper-parameter optimisation was performed for each MIMO size by training the models on known transmitted data at the identified optimal launch power. We maintained consistent filter lengths across all channels and steps to simplify the optimisation process.

For the L-simIVSTF, which was implemented with $N = 1$ [16], we optimised the length of the CD FIR filters S_{CD} in both the linear and nonlinear paths of the architecture, as well as the length of the fractional delay filters $S_{\text{fd}} = T$. Figure 6.5(a) shows the results of the CD FIR filter optimization, indicating that the model's performance improves with increasing filter length for lengths up to approximately 41 taps, beyond which it levels off.

A minimum bound for S_{CD} can be estimated by calculating the group delay difference induced by CD, which is expressed in number of samples as $T_{\text{CD}} = 2\pi\,|\beta_2|\,LB/T_s$ [58]. Assuming the bandwidth $B = (1+0.1)\cdot 32\,\text{GHz}$, the sampling interval $T_s = 1/(2\cdot 32\,\text{GHz})$,

and the fibre length covered by a single step $L = 100\,\text{km}$, $T_{\text{CD}} \approx 31\,\text{samples}$. This value aligns with the filter length at which performance begins to saturate, consistently with the behaviour observed for LDBP in Fig. 6 of [58]. While the optimal S_{CD} value found exceeds this bound, joint filter optimisation allowed the model to approach it closely. To avoid any performance penalties, $S_{\text{CD}} = 43$ taps was used for the results presented hereafter. Contrarily, as evidenced by Fig. 6.5(b), a length of just 4 taps is sufficient for the fractional delay filters to achieve optimum performance. Notably, the performance trends for both filter types are consistent across all MIMO sizes. Panels (b) and (c) of Fig. 6.5 provide further insights into the design of the fractional delay filters by showing the filter's magnitude and group delay, respectively, for different values of S_{fd}. The frequency response of an ideal delay system is given by $H_d(\omega) = \exp(-j\omega\tau_d)$, where τ_d is the delay in number of samples, corresponding to an all-pass filter with unit magnitude, $|H_d(\omega)| = 1$, and a constant group delay of $-\frac{d}{d\omega}\arg[H_d(\omega)] = \tau_d$. However, the FIR filter approximation deviates from this ideal response, with the extent of deviation depending on the type and size of the windowing function used. In our case, selecting a Hamming window effectively reduces the transfer function ripples associated with the Gibbs phenomenon. The flattest pass-band performance is achieved for filter lengths of more than 12 taps. While at shorter S_{fd} the transfer function deviates more from the optimal response, Fig. 6.5(b) highlights that as long as S_{fd} exceeds 4 taps, the induced discrepancies are effectively counterbalanced by the adaptive CD FIR filters. A length of $S_{\text{fd}} = 8$ taps was selected in the remainder of this paper. For the SPM and XPM filters in the L-simIVSTF, we assumed zero-valued initial conditions for all taps but the central ones, $\mu_{m,0}^{(q,n)}$ and $\nu_{m,p,0}^{(q,n)}$ which were set to initialisation factors ξ_{SPM} and ξ_{XPM}, respectively. We studied the role of these factors in the convergence of the model. The validation curves from the training process of a 5×5, one-step-per-span implementation are depicted in panels (a) and (b) of Fig. 6.6. These curves represent the evolution of the MSE over the number of training epochs for the validation dataset. We can observe from Fig. 6.6 (a) that the initialisation factor for the SPM filters has minimal influence on the convergence speed. By contrast, suitable initialisation of the XPM filters leads to a significantly faster convergence (Fig. 6.6 (b)). This is because XPM is the dominant effect responsible for transmission performance degradation. Repeating the same exercise for two-, three-, and four-steps-per-span configurations showed similar results. Optimising the initial conditions for the SPM and XPM filters facilitated the convergence of the model,

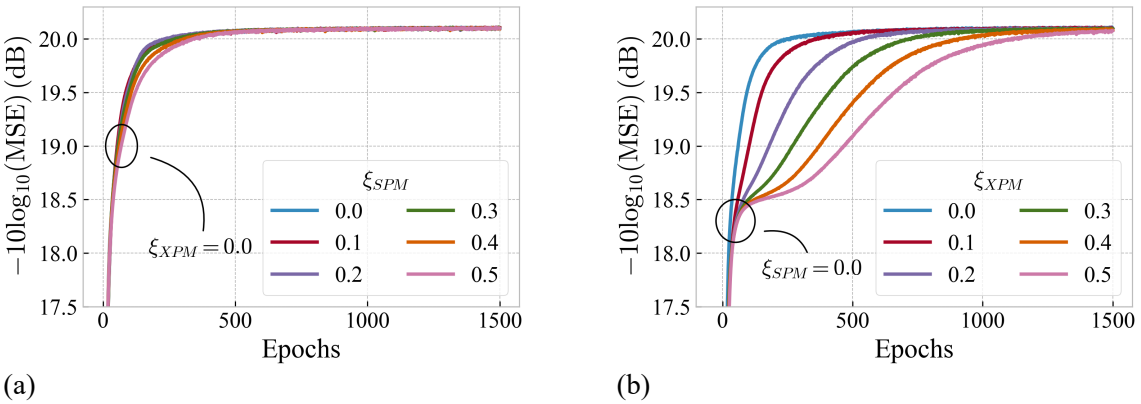


Figure 6.6: Optimisation of the initialisation factors for the SPM and XPM filters for a 5×5 L-simIVSTF model. The figures show the MSE evolution over the number of training epochs for varying (a) ξ_{SPM} and (b) ξ_{XPM} values.

thereby enabling the implementation of 5×5, 7×7 and 9×9 MIMO configurations at only one step per span. Furthermore, these results indicate that setting factors $\xi_{\text{SPM}}, \xi_{\text{XPM}}$ to zero is the best initialisation strategy. This initialisation approach has also been adopted for L-IVSTF and FE L-IVSTF models and employed in all subsequent results.

Next, we assessed the impact of the SPM and XPM filter lengths, S_{SPM} and S_{XPM}, on the equalisation capability of the L-IVSTF and L-simIVSTF models. The L-IVSTF scheme was implemented using $N = 4$, which is required to achieve comparable performance to the L-simIVSTF one [14]. The results are summarised in Fig. 6.7, indicating that whilst the performance of the L-IVSTF model is nearly insensitive to S_{SPM}, short SPM filters can enhance the performance of the L-simIVSTF model to a certain extent (panel (a)). This is attributed to the joint training of the SPM and CD filters in the L-simIVSTF model, which enables a more accurate approximation of the intra-channel impairments and improves the model's convergence. A saturation trend is observed, however, with filter lengths beyond 7 taps not yielding further performance gains. Therefore, we chose $S_{\text{SPM}} = 7$ taps as the optimal length. Regarding the XPM filters (panel (b)), the performance improves with increasing filter size consistently across both the L-IVSTF and L-simIVSTF models, while the saturation point shifts to a higher value as the MIMO size increases. A 41-tap XPM filter is sufficient for optimal performance; hence, it was used for all other results. It is worth noting that the need for XPM filters longer than the SPM ones is due to the channel walk-off effect, which extends the memory of the nonlinear interactions between co-propagating symbol streams across different wavelengths. Nevertheless, the same length

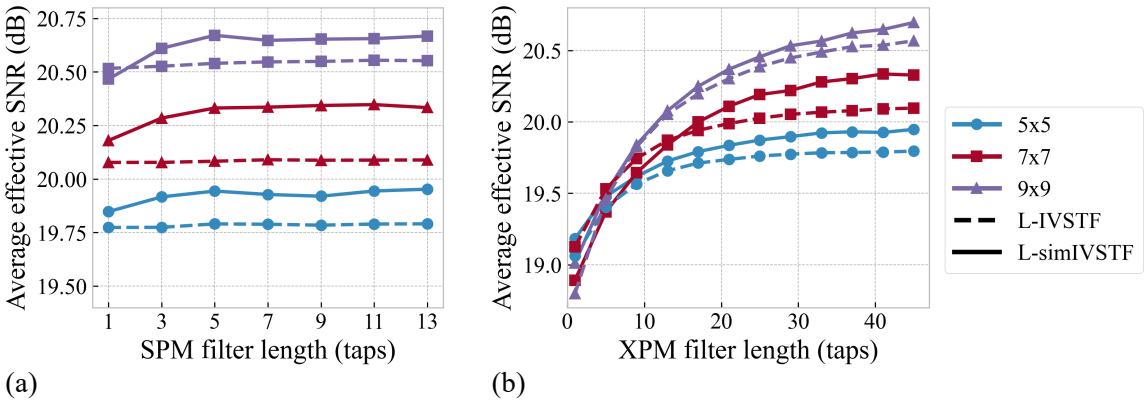


Figure 6.7: SPM and XPM filter length optimisation for the L-IVSTF and L-simIVSTF models. SNR performance when varying (a) SPM filter lengths using 41-tap XPM filters, and (d) varying XPM filter lengths using 7-tap SPM filters. SPM filtering benefits only the L-IVSTF model, while XPM filtering improves both L-IVSTF and L-simIVSTF models.

may not be necessary for all XPM filters within the structure, but can vary depending on the specific channel pairs considered in the MIMO LDBP model [124]. Therefore, while employing a uniform filter length—as done in this work—simplifies the design of the MIMO NLE architecture, tailoring the filter length to specific channel interactions could provide additional benefits by reducing the model's computational complexity.

We analysed the relationship between steps per span, performance, and convergence for the L-simIVSTF. Figure 6.8 (a) compares the convergence of a 5×5 model with and without optimised initialisation factors, ξ_{SPM} and ξ_{XPM}, across various steps per span configurations. For the unoptimised cases, the factors were set to 1. The results indicate that the algorithm converges to acceptable performance regardless of whether the factors are optimised. However, consistent with the results in Fig. 6.6, factor optimisation significantly affects convergence speed, irrespective of the step-per-span implementation. While increasing the number of steps per span also influences convergence speed, its effect depends on the initialisation strategy. With unity factors, increasing the steps has minimal effect, with all models requiring ~ 1500 epochs to converge. In contrast, when $\xi_{\text{SPM}} = \xi_{\text{XPM}} = 0$, convergence accelerates to ~ 250 epochs when using a 2 StpS configuration. Nevertheless, this improvement does not justify the additional complexity of increasing the steps. Therefore, a 1 StpS L-simIVSTF configuration is sufficient when appropriately initialised and trained long enough to ensure optimal performance.

Subsequently, we proceeded with the design of the linear stages for the L-IVSTF model, which are responsible for compensating CD effects. This equalisation is performed in a

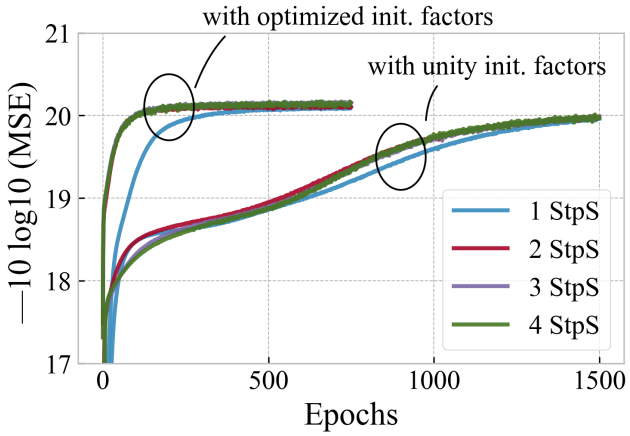


Figure 6.8: Impact of initialisation factors and number of steps per span on the convergence of the 5×5 L-simIVSTF model. The plot depicts the MSE evolution over the number of training epochs for unity-valued and optimised factors.

block-wise manner in the frequency domain and involves FFT/IFFT operations. To mitigate ISI at the edges of each block caused by the equivalent cyclic convolutions of the FFT/IFFT processes, we adopted the overlap-and-save method [101]. The overlap length is determined by the impulse response of the dispersed channel [142] and characterises the computational efficiency of the algorithm.

While there are estimates for the overlap length for single-channel operation [79], their extension to our MIMO L-IVSTF configuration would require accounting for the walk-off effects relative to the central wavelength of the equalisation band. This adjustment offsets the group delay of the associated filters by a term proportional to the channel index m, thereby resulting in an overlap length dictated by the walk-off between the outermost channels. Figure 6.9(a) illustrates the optimisation of the overlap and FFT-block lengths for the 7×7 MIMO implementation, indicating that a minimum N_e of 2048 samples is necessary to avoid inter-block interference [79]. If the overlap length is too short, the MIMO model may inadequately compensate for the channels at the edges. The block length N_{FFT}, set as a power of two to use the radix-2 FFT algorithm, is twice N_e, and further increase of its value showed negligible impact on performance. The overlap and block length pairs that we selected are (1024, 2048) for the 5×5 implementation, and (2048, 4096) for the 7×7 and 9×9 realisations.

Figure 6.9(b) shows the dependence of the model's performance on the number of steps per span for the different MIMO sizes (dashed curves). We can see that increasing N up to

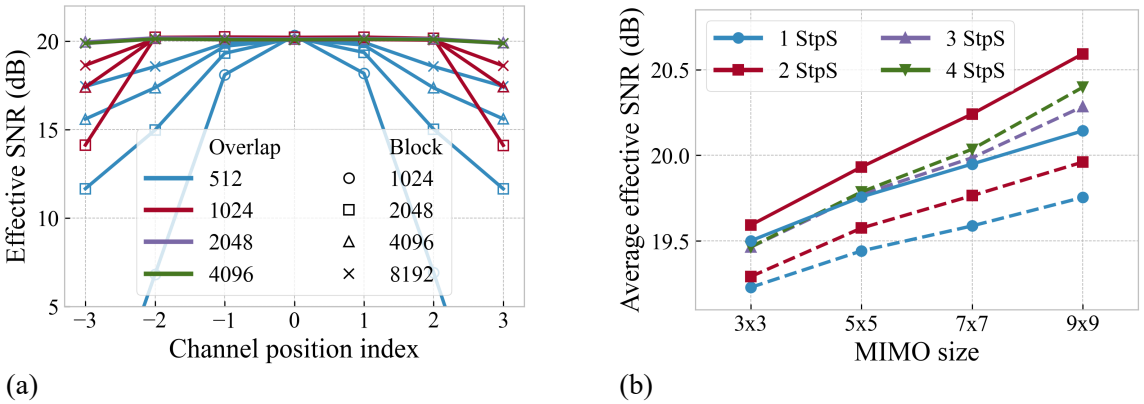


Figure 6.9: L-IVSTF model's performance in terms of (a) per-channel effective SNR for a 7×7 implementation with different overlap and block lengths in samples. An overlap size of 2048 samples is required to equalise all channels. (b) Average effective SNR for different step-per-span implementations. The FE L-IVSTF provides a higher performance than L-IVSTF, using fewer steps per span.

a certain extent consistently enhances performance, with the larger MIMO sizes benefiting more from such an increase. This trend results from the use of trainable filters in the nonlinear equalisation steps of the algorithm, as opposed to a standard (non-learned) IVSTF structure, which does not show such a performance gain [88]. The chosen $N = 4$ corresponds to a close-to-saturated value beyond which significant performance improvement is no longer attained.

Finally, we optimized the FE L-IVSTF architecture. While the lengths of the SPM and XPM FIR filters in the nonlinear stages were kept unchanged, we thoroughly studied the dimensioning of the trainable FIR CD filters, which depends on the amount of residual dispersion to be compensated for. Figure 6.10 shows the average effective SNR performance for the 7×7 MIMO implementation at 1 and 2 steps per span as a function of the filter size for different amounts of residual dispersion. For small residual dispersion, such as 17 ps/nm or less, a 7-tap filter is sufficient to provide optimum performance. Using shorter S_{CD} results in a performance penalty, which becomes more pronounced with increasing residual dispersion. Therefore, $S_{\mathrm{CD}} = 7$ taps was chosen as the optimum. We can also see in Fig. 6.10 that operating the algorithm at $N = 2$ brings about higher equalisation performance compared to the $N = 1$ operation, as it should be expected. The comparison with L-IVSTF model provided in Fig. 6.9(b) shows that the FE L-IVSTF model operated at $N = 1$ matches the performance of its non-FE counterpart at $N = 4$ for the 3×3 and 5×5 MIMO implementations. At the same time, operating the FE L-IVSTF model at

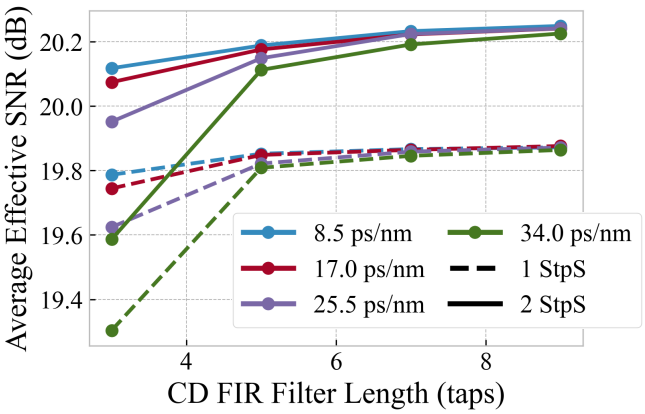


Figure 6.10: Average effective SNR performance of the 7×7 FE L-IVSTF model at 1 and 2 steps per span (dashed and solid curves, respectively) versus FIR filter length for different combinations of adaptive dispersion. Close-to-optimal performance is achieved by addressing an 8.5 ps/nm dispersion with a 7-tap trainable CD filter.

$N = 2$ surpasses the performance of any L-IVSTF realisations. Therefore, we chose $N = 2$ as the optimal number of steps per span.

The hyper-parameter tuning described above allowed us to inspect the behaviour of each algorithm and identify their optimum settings. Figs. 6.11 and 6.12 illustrate one of the major results of this thesis, i.e., a head-to-head comparison of the NLE performances of the different optimised models. Fig. 6.11 shows the average effective SNR as a function of the per-channel launch power. Also shown is the performance curve for linear CD equalisation (green). We can see that the SNR improvement over CD equalisation enabled by our schemes ranges between ~ 1.2 dB (5×5 L-IVSTF) and ~ 2.2 dB (9×9 L-simIVSTF and FE L-IVSTF) at the respective optimum launch powers. To our knowledge, the L-simIVSTF model offers the largest performance improvement over CD equalisation achieved by single-step-per-span MIMO models in theoretical studies. In contrast, a LDBP counterpart demonstrated only a 1.3 dB Q^2-factor improvement by applying a 5×5 scheme to an 11-WDM channel 40×80 km transmission[124]. The L-simIVSTF and FE L-IVSTF models outperform the L-IVSTF scheme and feature equal performance at the optimum launch power across all MIMO sizes, while the L-simIVSTF model is more tolerant than the FE L-IVSTF scheme to powers beyond the optimum one.

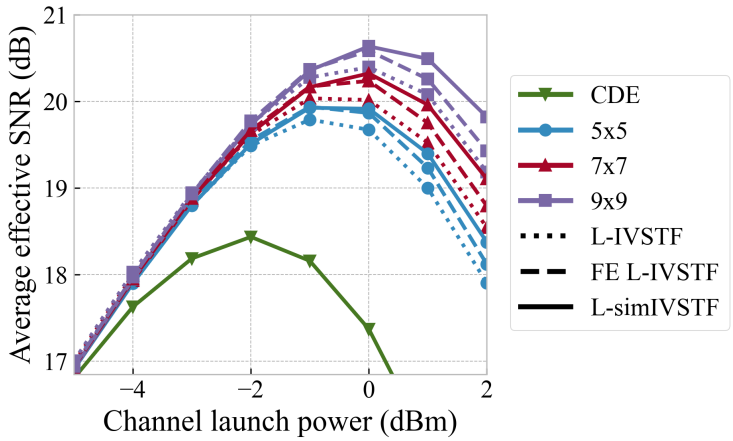


Figure 6.11: Average SNR against channel launch power for L-IVSTF, FE L-IVSTF and L-simIVSTF MIMO models implemented with 4, 2 and 1 steps per span, respectively. L-simIVSTF demonstrates the best performance across MIMO sizes.

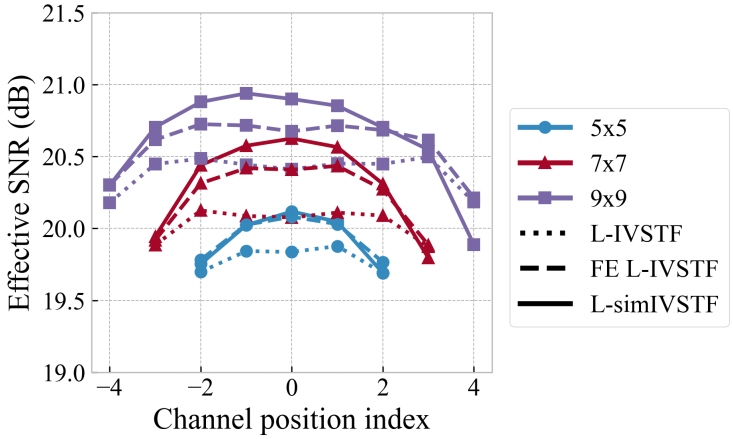


Figure 6.12: SNR at each channel position for L-IVSTF, FE L-IVSTF and LsimIVSTF, using 4, 2 and 1 steps per span, respectively. Channels near the centre benefit the most in all cases. Larger MIMO sizes increase the number of near-optimal channels, and improve uniformity across the bandwidth.

The SNR performance for each individual channel at the optimum launch power is depicted in Fig. 6.12. It is important to note that only a subset of the 11 wavelength channels in our data transmission band, defined by the MIMO order, is equalised by the IVSTF algorithms. Consequently, the central channel gets the most benefit, while the channels at the band edges are only partially equalised. Furthermore, the L-simIVSTF model has higher adaptability than the FD models, allowing it to better account for the effects of inter-channel nonlinearity and enabling a more effective performance enhancement for the central channels during the optimisation process. Consequently, the central channels improve earlier during training. As the loss function minimises the average loss across all

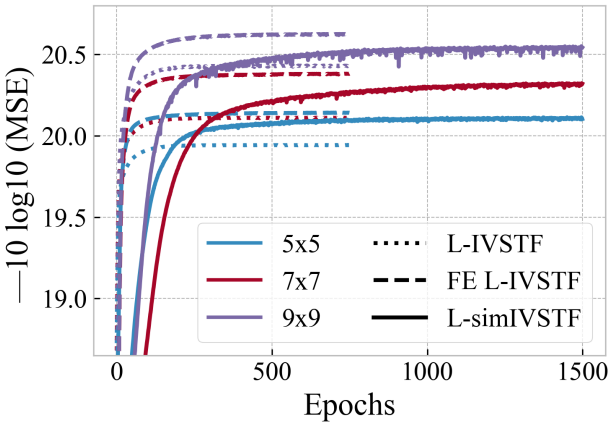


Figure 6.13: Convergence performance of the various MIMO models. The L-simIVSTF model takes significantly longer to converge than the L-IVSTF and FE L-IVSTF ones.

channels without prioritising specific ones, the early improvements in central channels cause a local minimum during optimisation, resulting in lower performance levels for the edge channels.

Figure 6.13 gives insight into the convergence behaviour of the different algorithms by showing the average effective SNR, derived from the MSE, of the validation data set, as a function of the number of training epochs. It is seen that both FD models achieve convergence within 300 epochs regardless of their MIMO size. Conversely, the L-simIVSTF model exhibits a slower convergence rate, attributable to its considerably larger number of trainable parameters. Moreover, unlike the FD models, its convergence rate decreases as the MIMO size increases because an increase in the number of processed channels causes a sharp increase in trainable parameter count.

6.4.1 Generalisation

We investigated the generalisation capabilities of our schemes. We begin by examining the constellations of the equalised signals. Figure 6.14 shows the constellation diagrams of the centre channel signals equalised using CDE and FE L-IVSTF and L-simIVSTF models of size 9×9. The constellations in Figs.(b) and (c) do not exhibit signs of overfitting, and show visibly reduced nonlinear distortion, consistent with the improvements in average effective SNR shown in Figures 6.11 and 6.12. In contrast, black-box NN models are prone to overfitting, which typically appears in the constellation diagrams as a jail window-like pattern. No such patterns were observed in the outputs of any of our schemes. The resilience

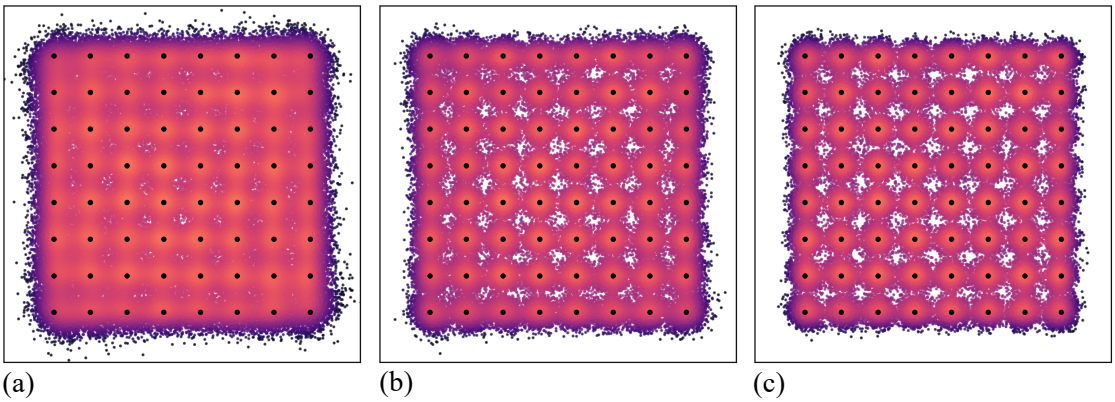


Figure 6.14: Constellation diagrams of the centre channel signals equalised with (a) CDE and (b) FE L-IVSTF, and (c) L-simIVSTF 9×9 models. There are no signs of the jail window pattern typical of black-box models.

of the L-IVSTF schemes to overfitting may be attributed to their relatively small parameter space compared to the amount of training data. Moreover, they are initialized with weights derived from the link parameters, placing them close to the optimal solution.

We also assessed the model's ability to generalise across varying operating conditions, examining whether the weights learned at one launch power could be applied to a different launch power. The results, shown in Figure 6.15, correspond to the 5×5 2 StpS L-simIVSTF model presented in [13], applied to a 5-channel WDM Tx scenario considering a 1000 km link. The light blue curve shows the performance of models trained individually for each launch power. In this case, optimal performance occurs at 1 dBm. The remaining curves represent models trained on a specific launch power but applied to data from launch powers on which they were not trained. Two trends are observed. Models generalise well beyond the training launch power for launch powers below -1 dBm, where system operation is linear. Conversely, for -1 dBm and above, the performance of models applied to scenarios they were not trained on is suboptimal. We conclude that the weights learned in the nonlinear regime are launch-power dependent. Therefore, separate models must be trained for each launch power to ensure optimal performance.

To better understand the effectiveness of the learned Volterra models, we analysed their learned parameters. Unlike in conventional deep neural networks, where the role of each layer is often unclear, each step of our algorithms has a clearly defined function: the linear steps apply dispersion and time delay, while the nonlinear steps introduce instantaneous phase shifts. While well-understood, the algorithm performs poorly when the nominal val-

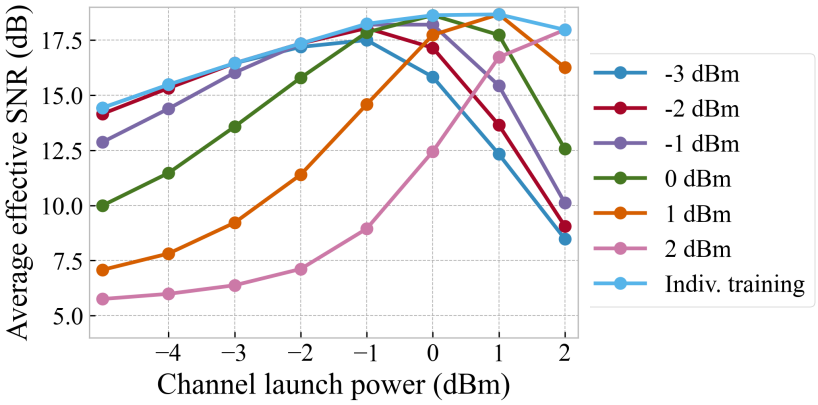


Figure 6.15: Generalisation performance of 2 steps-per-span 5×5 L-simIVSTF models across launch powers. Each model is trained on data corresponding to a specific launch power. The performance curve for models evaluated on data matching their training launch power is shown for reference. Models applied to data from different launch powers show degraded performance in at least some cases. Furthermore, models trained on high launch power exhibit particularly poor generalisation.

ues from analytical solutions [88, 136] are employed and joint optimisation is not performed. This is due to the coarse approximation of the fibre link response these parameters give. Machine learning optimisation has proven essential for enabling the model to accurately identify the parameters that enable inter-channel equalisation. What remains unclear is how the initial filter coefficients are adjusted to achieve an acceptable performance. Examining the learned parameters could offer insights into how the model compensates for imprecisions in the equalisation structure. Since the architecture is a network of linear filters and nonlinear operations, parameters can be analysed with the assistance of digital filter design theory. Specifically, we analyse the frequency response of learned filters. Previous studies for LDBP support this approach. For example, [58] analysed the response of the CD filters and observed that the optimisation process, rather than prioritise the response of any individual filters, optimises their combined response. We perform this analysis on the L-simIVSTF, where all linear and nonlinear steps are trainable. In this model, each channel processing unit m has two distinct linear CD filters sequences, which can be analysed separately. The horizontal or "**trunk**" filter sequence corresponds to the first-order kernel from the IVSTF and primarily compensates for the chromatic dispersion in the link. However, the filters in this sequence differ in function from the corresponding branch in the IVSTF scheme since they also account for the effect of dispersion on nonlinear phase shifts. Conversely, the filters in the "**branch**" sequence represent the location where the lumped nonlinear shifts occur in

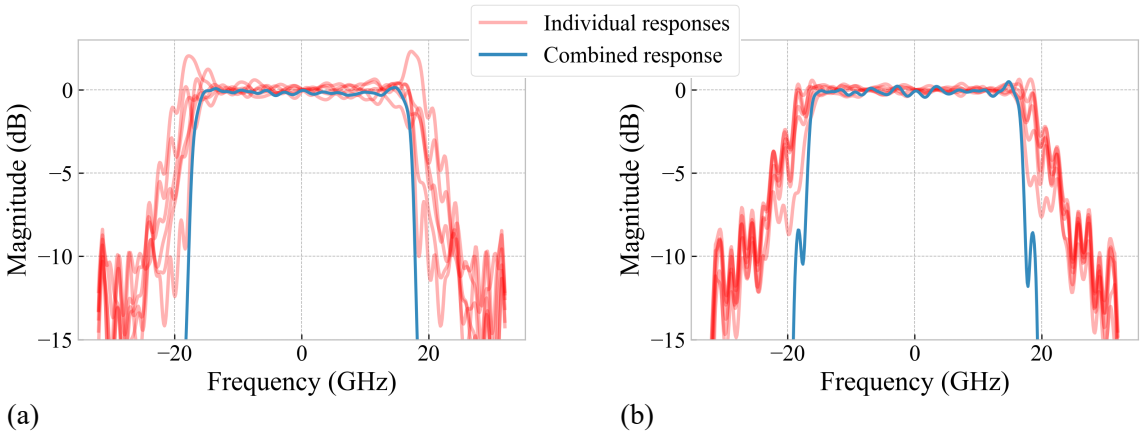


Figure 6.16: Individual and combined responses of the filter sequences corresponding to the (a) trunk and (b) nonlinear branches of the central channel of a 5×5 LsimIVSTF model.

the fibre link. While simIVSTF —through nonlinear branch interconnections— relaxes the precision requirements of individual filters, it also imposes its own precision requirements: The filter in branch i requires an adequate response from the filters of preceding branches $1, 2, ..i-1$ to provide an accurate estimation of the nonlinear phase shift of branch i.

We analyse the learned solutions of a fully time-domain model: a 5×5 L-simIVSTF operated at 1 step per span. First, we inspect the CD filters in the linear steps. Figures 6.16 (a) and (b) present the individual and combined amplitude responses of the CD filters for the trunk and branch paths of the centre channel. The combined response is obtained by convolving the impulse responses of individual filters. For both paths, the learned CD filters significantly deviate from the ideal chromatic dispersion filter response, consistent with findings in [125]. These filters exhibit considerable out-of-band gain. Nevertheless, the combined responses for both signal paths approximate the ideal CD filter. The out-of-band ripples cancel each other, resulting in a near-constant amplitude across the frequency range of the equalised signal (32 GHz). Next, we consider the filters in the nonlinear steps. Figure 6.17 (a) shows the real-valued taps of the SPM filters for the centre channel across the model's steps. The SPM filters exhibit a triangular window shape, similar to those reported in [124], which applies a linearly decaying weighting to the power of adjacent samples.

Figure 6.17 (b) illustrates the XPM filter taps across the model's steps. Due to the large number of XPM filters in the structure, we present only those relating the centre channel (index 0) and an edge channel (index $i = -1$) for brevity. The responses of these filters are asymmetric, with their shapes varying based on the channel pair. Filters processing the

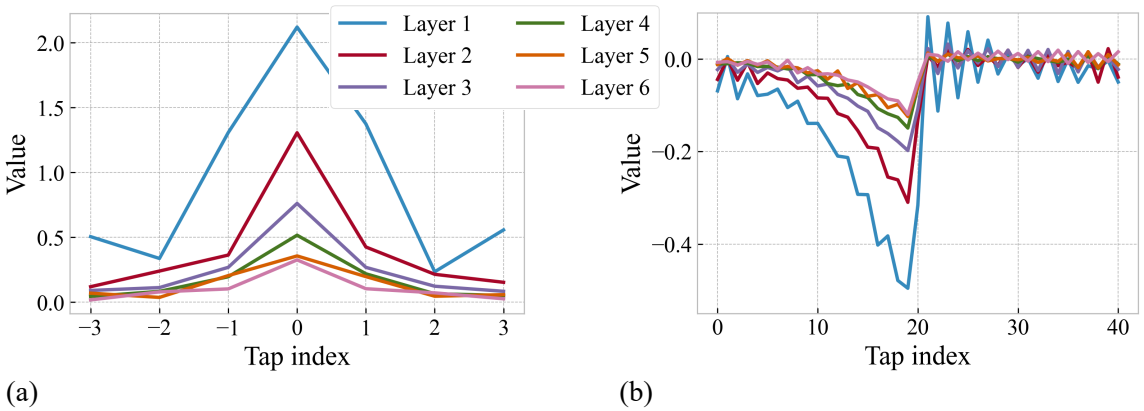


Figure 6.17: For a 1 StpS 5 × 5 L-simIVSTF model: Coefficient values of (a) the SPM filters corresponding to the centre channel and (b) the XPM filters processing contributions to the centre channel from first channel to the left.

nonlinear shift induced by adjacent channels on the centre channel show a clear inflexion point at the centre tap, with peaks shifted left or right depending on the position of the influencing channel. Finally, the coefficient amplitudes of the SPM and XPM filters decrease depending on how deep a layer is located in the model. This indicates that the model prioritises the contributions from the initial layers, progressively reducing the weight of filters in deeper layers.

6.4.2 Complexity Analysis

To fully appraise the various Volterra-based NLE schemes, in this section we perform a complexity analysis using the most expensive computations, namely, the required number of real multiplications per transmitted symbol (RM/sym), while neglecting the less costly addition operations [42].

Starting with the FD operations involved in the L-IVSTF model, the signal is converted to the FD at the beginning of the process and only reverted to the TD before each nonlinear stage to economise on the FFT/iFFT transformations within the structure, and final conversion to the TD is done at the structure's output (Fig. 6.3 (b). Therefore, the number of FFT/iFFT pairs needed is $N_s + 1$, where $N_s = NQ$ is the total number of steps. With a radix-2 implementation, each pair has a cost of $C_{\text{FFT}} = 4N_{\text{FFT}} \log_2(N_{\text{FFT}})$ RMs for a block of N_{FFT} samples. The number of H_m transfer functions in the m-th channel unit is $2N_s + 1$, each incurring a cost of $4N_{\text{FFT}}$ RMs from their element-wise complex multiplication with the transformed signal. The overlap-and-save block processing required for the operation of

the model increases the computational complexity due to the overlapping samples, where the minimum required overlap length N_e increases with the number of processed channels M, as observed in Sec. 6.4. Hence, the total computational cost of the linear FD filtering is given by

$$C_{\mathrm{FD}} = \frac{p(N_s+1)C_{\mathrm{FFT}} + (4pN_{\mathrm{FFT}})(2N_s+1)}{N_{\mathrm{FFT}} - N_e + 1} \text{ [RM/sym]}, \tag{6.15}$$

where p is the digital sampling rate. Concerning the TD operations in the nonlinear stages of the L-IVSTF architecture, the nonlinear activation functions cost $4p$ RM/sym due to squared signal modules and multiplications by complex constants. The filtering of signal powers inside the channel units entails 1 SPM and $M-1$ XPM operations per span. The SPM filters yield a cost of pS_{SPM} RM/sym while the XPM filters result in a cost of pS_{XPM} RM/sym. Therefore, the total cost of the TD operations is

$$C_{\mathrm{TD}} = pN_s(S_{\mathrm{SPM}} + (M-1)S_{\mathrm{XPM}} + 4) \text{ [RM/sym]}, \tag{6.16}$$

yielding a total complexity of $C_{\mathrm{L-IVSTF}} = C_{\mathrm{TD}} + C_{\mathrm{FD}}$. For the FE L-IVSTF model, the cost of the FD operations per step does not change, but the per-step cost of the TD operations is higher due to the incorporation of the short CD FIR filters. The convolution of a complex-valued signal with a complex-valued filter of length S_{CD} costs $4pS_{\mathrm{CD}}$ RM/sym [124]. Two of these convolutions are required per channel and per step. Therefore, the complexity of the FE L-IVSTF model, $C_{\mathrm{FEL-IVSTF}}$, exceeds $C_{\mathrm{L-IVSTF}}$ by $8pMN_sS_{\mathrm{CD}}$ RM/sym.

Finally, we consider the L-simIVSTF scheme. In each channel unit there are $2N_s$ linear steps, each requiring convolving the signal with a CD FIR filter of length S_{CD} RM/sym. Fractional delay filtering adds a convolution between a complex-valued signal and a real-valued filter, which costs $2pS_{\mathrm{fd}}$ RM/sym. Circular shifting of the signal is also performed within the step, but its cost is neglected in this analysis. The cost contribution of the nonlinear stages is the same as that of the L-IVSTF model. Therefore, the total complexity of the model is

$$C_{\mathrm{L-simIVSTF}} = pN_s(8S_{\mathrm{CD}} + 4S_{\mathrm{fd}} + S_{\mathrm{SPM}} + (M-1)S_{\mathrm{XPM}} + 4) \text{ [RM/sym]}. \tag{6.17}$$

For comparison, we also consider the complexity of single-channel DBP [98]:

$$C_{\text{DBP-1ch}} = 4pN_s \left(\frac{N_{\text{FFT}}(\log_2 N_{\text{FFT}} + 1)}{(N_{\text{FFT}} - N_e + 1)} + 1 \right) \text{ [RM/sym]}. \tag{6.18}$$

Based on the calculations above and the results of Sec. 6.4, we now examine each architecture's opportunities to achieve favourable performance-complexity trade-offs. First, we consider the per-step complexity of the FD schemes. We can see from Eq. (6.15) that optimising the FFT-block size N_{FFT} has little impact on reducing the overall step cost. This is because the optimised placement of FFTs in the schemes makes linear filtering relatively inexpensive. Conversely, XPM filtering is the most expensive computation of each step due to the large XPM filters required (Eq. (6.16)), with the number of operations scaling as $M - 1$. Therefore, as mentioned in Sec. 6.4, a promising strategy to reduce complexity would be optimising the XPM filter length for each individual channel. An inspection of Eqs. (6.15) and (6.16) reveals that the number of steps per span is the dominant factor in the overall complexity. Consequently, since the FE L-IVSTF model requires fewer steps per span than the L-IVSTF one to achieve optimal performance, as seen in Fig. 6.9(b), it is computationally more efficient. The results of Fig. 6.9(b) also highlight that better trade-offs can be attained for the FE L-IVSTF model with small performance compromises. For example, for the 9×9 MIMO implementation, accepting a 2% performance reduction enables halving the complexity by using $N = 1$, which still would yield ~ 1.7 dB improvement over CD equalization (Fig. 6.11).

In the L-simIVSTF scheme, all filtering operations employ convolutions. Their impact on complexity varies depending on the filter type: the CD filters are expensive due to their complex-valued taps, while the SPM, XPM and fractional delay filters are less costly as they use real-valued taps. The required number of each type of filter also differs, where more XPM than CD, fractional or SPM filters are needed for the MIMO sizes being considered. Figure 6.18 illustrates the relative impact of the SPM, XPM and CD filters on the total complexity by showing how $C_{\text{L-simIVSTF}}$ varies as a function of the length of each filter type with the other filter lengths fixed at their optimal values given in Sec. 6.4, for the 7×7 MIMO implementation. The fractional delay filters are excluded from this analysis due to their minimal contribution. We can see that changing S_{SPM} has negligible impact on com-

plexity (red line). While the XPM filters have a larger impact on complexity as evidenced by the steeper slope of the corresponding line (purple), it is the CD filters that affect complexity the most (blue line). Yet, at their optimal lengths, CD and XPM filters contribute almost equally to the overall complexity, as shown by the convergence of the corresponding lines. This scenario would change for larger MIMO sizes, where XPM filters dominate the computational cost. As with the FD models, computational efficiency improvements could be achieved by optimising S_{XPM} on a per-channel basis.

A head-to-head comparison of the complexity of the optimised schemes shown in Fig. 6.11 is presented in Fig. 6.19, constituting another key contribution of this thesis.

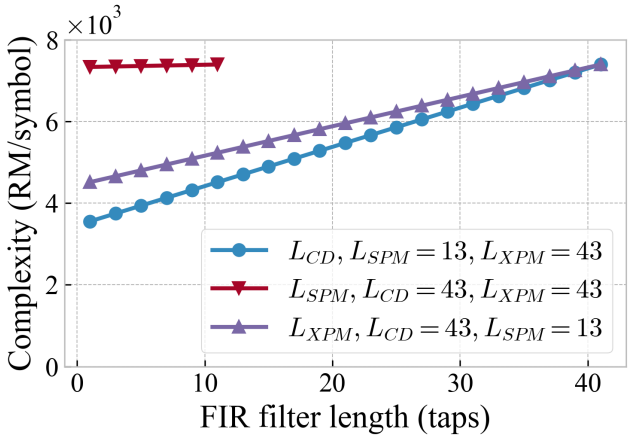


Figure 6.18: Complexity as a function of the length of various filter types in the L-simIVSTF model. Each curve is taken by varying the length of a filter type while keeping the length of other filter types fixed to their optimal values. The CD and XPM filters have a similar impact on complexity.

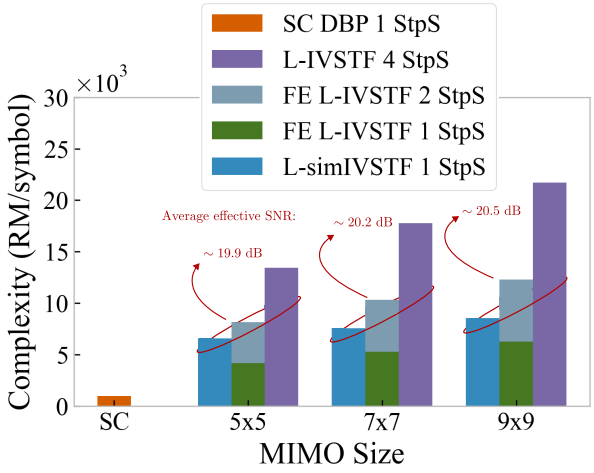


Figure 6.19: Complexity of best-performing models. The FE L-IVSTF 1 StpS model exhibits the lowest cost.

While the L-IVSTF model features the lowest per-step complexity, its need to operate at $N = 4$ results in the highest total complexity across all MIMO sizes. The FE L-IVSTF scheme at $N = 2$ follows up in complexity for the larger MIMO sizes, with a drastic complexity reduction resulting from requiring a halved number of steps per span for optimal performance. Furthermore, as discussed above, 1 step-per-span operation is viable for this model since it results in a small performance penalty. This halves the cost, thereby leading to the lowest complexity across all the MIMO sizes. The L-simIVSTF model has a comparable cost to the 2 step-per-span FE L-IVSTF scheme. Despite being able to operate at $N = 1$, it has the highest per-step cost due to the large CD filters required. We can therefore conclude that the FE L-IVSTF model operated at $N = 1$ affords the best performance-complexity trade-off: while it delivers $\sim 20\%$ less maximum improvement over CD equalisation than its 2 step-per-span counterpart or the L-simIVSTF model (Fig. 6.11), it requires only around half their cost.

6.5 Conclusions

We have presented a unified overview of different MIMO Volterra-based NLE schemes for WDM transmission systems enabled by ML. We have provided the mathematical foundations of the NLE models, described the optimisation of their hyper-parameters, assessed numerically their equalisation performance and quantified their computational complexity. The L-simIVSTF model, a simplified TD architecture relying on efficient FIR filter re-use in the linear stages and enhanced filtering in the nonlinear stages, achieves robust performance at only 1 step per span. This is due to its large flexibility, which stems from the adaptability of both linear and nonlinear stages. The FE L-IVSTF model, an FD structure with adaptive nonlinear stages enhanced by filtering both the power and optical signal waveforms, can attain the same performance at a similar computational cost when operating at 2 steps per span. Both models have been shown to afford an average SNR gain of $\sim 2.2\,\mathrm{dB}$ over CD equalisation for a 9×9 MIMO implementation. Operation of the FE L-IVSTF model at 1 step per span requires approximately half the cost of the L-simIVSTF scheme or its 2-step-per-span version at the expense of $\sim 20\%$ less SNR improvement over CD equalisation. Therefore, the FE L-IVSTF model appears to be the one that offers the best performance-complexity trade-off. Future work will look at further improvements of the FE

L-IVSTF architecture by optimising each MIMO channel path's configuration separately. The analysis and results presented in this chapter provide useful guidelines for the practical design of adaptive and low-latency multi-channel equalisers.

Chapter 7

Conclusion

7.1 Summary of Contributions

Contributions to Single-channel Equalisation

- Developed a learned time-domain model, **L-simIVSTF**, which showed a 3 dB SNR improvement over CDE in a 10×100 km scenario. We demonstrated that the gradient-based optimisation of the Volterra-based equalisation structure can overcome its accuracy limitations, matching the performance of its 1 StpS LDBP counterpart.

Contributions to Multi-channel Equalisation

- Developed a modular framework for implementing and training model-driven MIMO equalisers, which allows researchers to assemble multichannel equalisation schemes efficiently. The framework supports the investigation of model-driven equalisation architectures and the further development of MIMO-based DSP. Plans are in place to release the codebase as open source.

- Introduced **L-IVSTF**, the first MIMO-WDM approach based on extending and optimising the IVSTF using gradient-based techniques, enabling equalisation of inter-channel impairments in WDM transmission systems.

- Extended the single-channel time-domain **L-simIVSTF** for MIMO operation. This model offers the highest performance improvement among our MIMO equalisers: 2.3 dB improvement over CDE in an 11-channel 6×100 km transmission scenario.

- Proposed the **field-enhanced L-IVSTF** (**FE L-IVSTF**), which outperforms **L-IVSTF** while halving the computational steps required for effective nonlinearity mitigation. The novel filtering strategy employed in the FE L-simIVSTF, integrating static and trainable linear stages to balance adaptability and computational efficiency, could be applied to other nonlinear equalisation schemes. A comprehensive comparison of all proposed equalisation schemes identifies the **FE L-IVSTF** as the most efficient option in terms of performance and computational cost.

7.2 Limitations and Future Work

During this project, several opportunities for further research were identified:

- **Experimental Validation**

  The equalisation approaches presented in Chapters 5 and 6 have not yet been validated experimentally. Several technical challenges need to be addressed. Experimentally validating our single-channel scheme requires integrating our training pipeline with the DSP required to process experimental data. In addition, our multichannel schemes require simultaneous detection of transmitted channels. Alternatively, sequential channel detection may be performed, which requires solving synchronisation issues.

- **Investigating Alternative Structures**

  The equalisers we have developed are based on the simplification and enhancement of the IVSTF model [88]. Although the IVSTF topology leads to straightforward implementations and interpretable solutions, it restricts the possible topologies that can be obtained for efficient and trainable model-driven schemes. Exploring more general architectures with a potential for parameterisation, such as the VSNE [56], could yield other efficient and flexible structures with low computational costs.

- **Addressing Time-varying Impairments**

  Our study has not accounted for time-varying impairments in optical transmission systems, such as ADC nonlinearity or laser phase noise. Addressing these effects in our receiver architecture would require integrating the ML training pipeline with the adaptive DSP required for their mitigation. A potential approach that could be

followed is outlined in [36], where the adaptive DSP stages needed to compensate for these impairments are also deep-unfolded and integrated into the equalisation model.

- **Further Complexity Reductions**

  Finally, there are further opportunities to reduce the computational cost of the proposed models, as key findings from other models suggest. It has been shown that the complexity of learned DBP does not scale linearly with the number of steps since the length of the filters employed at each step can be reduced to the limit imposed by channel memory, unlike with conventional DBP [58]. Similar outcomes may be observed in our algorithms. The relationship between steps per span and complexity for time-domain Volterra models is yet to be adequately explored, and with it the possibility of low-complexity multi-span implementations. Another option is to investigate variable processing rates within our equalisers. The proponents of IVSTF employed lower processing rates for the nonlinear paths, where reducing the processing rate by half cuts the computational cost by the same factor, with only a minor sacrifice in performance [88]. The effect of such rate reduction strategies in our equalisers is yet to be evaluated.

- **Improved Complexity Estimations**

  Our evaluation of the computational cost of the algorithms proposed in this thesis is limited. We have employed the required number of real multiplications as a proxy for power dissipation, which is the ultimate measurement of computational cost. Therefore, cost may be underestimated. Moreover, other important implementation aspects, such as circuit design and the required ADC resolution, are yet to be studied.

Appendix A

Derivation of the Volterra Equaliser

In this chapter, we derive the inverse multichannel transfer function. We first consider a single backpropagation step of a field E_0 over a single span of fibre h_1 as shown in Figure A.1.

A linear step is defined as

$$E_1^{(l)} = F^{-1}\left\{F\{E_0\}H_1(\omega)\right\} \tag{A.1}$$

Here, the transfer function $H_1(\omega)$ characterises the dispersion of the fibre length h_1. By approximating the exponential of the nonlinear step of the SSF solution as a first order polynomial , the output field E_1 can be expressed in terms of $E_1^{(l)}$ as [136]

$$E_1 \approx E_1^{(l)} e^{\alpha h_1/2} + j\gamma E_1^{(l)} e^{\alpha h_1/2} |E_1^{(l)}|^2 h_{1,\text{eff}} \tag{A.2}$$

Here, $h_{1,\text{eff}} = e^{\alpha h_1}/\alpha$.

We now consider two backpropagation steps over the same fibre length. We define the second linear step as

$$E_2^{(l)} = F^{-1}\left\{F\{E_1\}H_2(\omega)\right\} \tag{A.3}$$

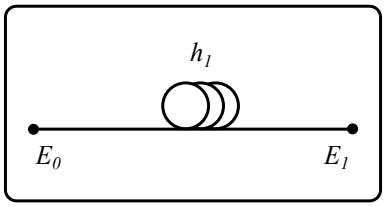


Figure A.1: Diagram of an inverse link of a single step h_1.

The output field E_2 is approximated by

$$E_2 \approx E_2^{(l)} e^{\alpha h_2/2} + j\gamma E_2^{(l)} e^{\alpha h_2/2} \left|E_2^{(l)}\right|^2 h_{2,\text{eff}} \tag{A.4}$$

Expanding the linear step $E_2^{(l)}$ results in the following sum of terms:

$$E_2^{(l)} = E_{2,A}^{(l)} + E_{2,B}^{(l)} \tag{A.5}$$

where

$$E_{2,A}^{(l)} = F^{-1}\left\{F(E_0)H_1(\omega)H_2(\omega)e^{\alpha h_1/2}\right\} \tag{A.6}$$

and

$$E_{2,B}^{(l)} = F^{-1}\left\{F\left\{j\gamma E_1^{(l)}\left|E_1^{(l)}\right|^2 e^{\alpha h_1/2} h_{\text{eff}}\right\} H_2(\omega)\right\} \tag{A.7}$$

Substituting A.6 and A.7 in A.4, the output field E_2 is finally expressed as

$$\begin{aligned} E_2 \approx{}& F^{-1}\left\{F(E_0)H_1(\omega)H_2(\omega)e^{\alpha(h_1+h_2)/2}\right\} \\ &+ F^{-1}\left\{F\left\{j\gamma E_1^{(l)}\left|E_1^{(l)}\right|^2 e^{\alpha h_1/2} h_{\text{eff}}\right\} H_2(\omega) e^{\alpha h_2/2}\right\} + j\gamma E_{2,A}^{(l)} e^{\alpha h_2/2}\left|E_{2,A}^{(l)}\right|^2 h_{\text{eff}} \end{aligned} \tag{A.8}$$

We now generalise to M steps over a single span of fibre. We first define $E_M^{(l)}$ as a recursive step in terms of the field at the output of step $M-1$:

$$E_M^{(l)} = E_{M-1} F^{-1}\left\{H_M(\omega)\right\} \tag{A.9}$$

Also, for convenience, we define a linear step $E_k^{(ll)}$ encompassing k steps:

$$E_k^{(ll)} = E_0 F^{-1}\left\{H_1(\omega)H_2(\omega)\ldots H_k(\omega)e^{\alpha/2(h_1+h_2+\ldots+h_{k-1})}\right\} \tag{A.10}$$

The output field after M steps E_M is

$$\begin{aligned} E_M = {}& E_0 F^{-1}\left\{\prod_{k=1}^{M} H_k(\omega) e^{\alpha h_k/2}\right\} \\ &+ j\gamma \sum_{k=1}^{M} E_k^{(ll)} e^{\alpha h_k/2}\left|E_k^{(ll)}\right|^2 h_{k,\text{eff}} F^{-1}\left\{\prod_{k=k}^{M} H_n(\omega) e^{\alpha h_k/2}\right\} \end{aligned} \tag{A.11}$$

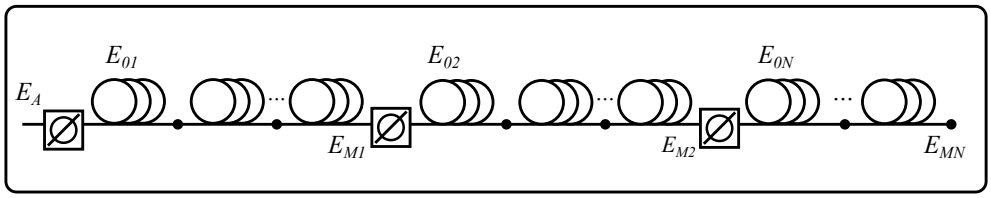


Figure A.2: Diagram of an inverse link of M spans, where each span is subdivided in N steps.

We now generalise for N steps and M spans, considering an input field E_A, as shown in Figure A.2. We assume identical lump losses $\sqrt{L}$ at the beginning of each span of fibre. The output field E_{MN} is given by

$$\begin{aligned} E_{MN} = E_A \left(\sqrt{L}\right)^N F^{-1}\left\{\prod_{k=1}^{M}\prod_{m=1}^{N} H^{(k,m)}(\omega) e^{\alpha h_k^{(m)}/2}\right\} \\ + j\gamma \sum_{k=1}^{M}\sum_{q=1}^{N} \left(\sqrt{L}\right)^{N-q} E_{kq}^{(ll)} e^{\alpha h_k^{(q)}/2} \left|E_{kq}^{(ll)}\right|^2 g_k^q \\ + F^{-1}\left\{\prod_{n=k}^{M} H^{(n,q)}(\omega) e^{\alpha h_k^{(q)}/2} \prod_{k=1}^{M}\prod_{m=q+1}^{N} H^{(k,m)}(\omega) e^{\alpha h_k^{(m)}/2}\right\} \end{aligned} \tag{A.12}$$

where $H^{(k,m)}(\omega)$ is the transfer function for span k and step m, and

$$E_{kq}^{(ll)} = E_A \left(\sqrt{L}\right)^q F^{-1}\left\{\prod_{k=1}^{M}\prod_{m=1}^{q-1} H^{(k,m)}(\omega) e^{\alpha h_k^{(m)}/2} \prod_{n=1}^{k-1} H^{(n,q)}(\omega) e^{\alpha h_n^{(q)}/2} H^{(k,q)}(\omega)\right\} \tag{A.13}$$

Appendix B

Diagrams of MIMO Volterra Architectures

This appendix presents additional diagrams of the MIMO Volterra architectures, providing alternatives to those in Figure 6.3, 6.4. These diagrams were previously included in our conference contributions [13, 14].

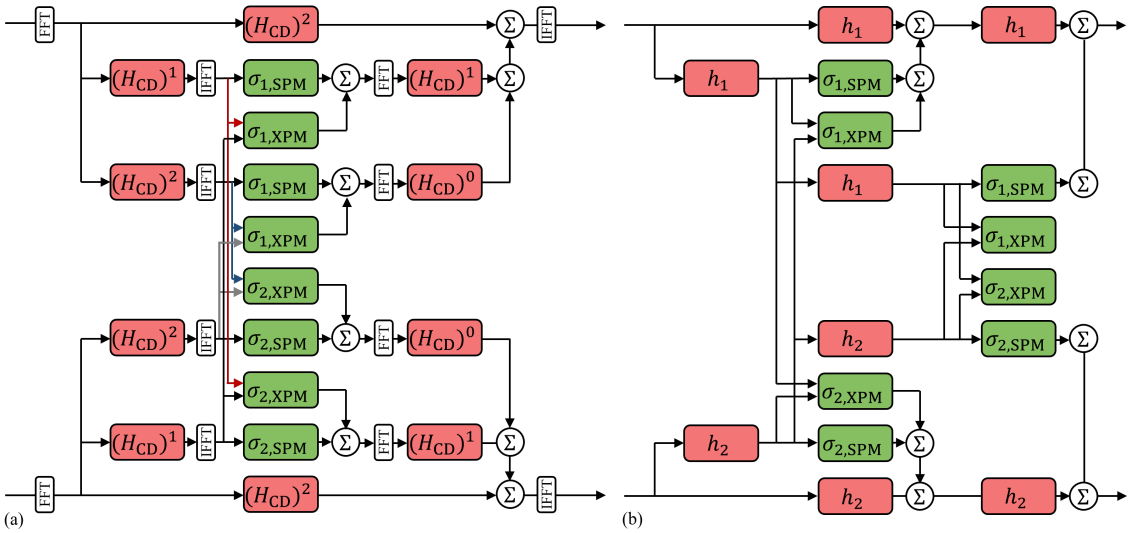


Figure B.1: Diagram of 2×2 MIMO schemes: (a) IVSTF, (b) simIVSTF.

Appendix C

Implementation of Volterra-based Equalisers

This appendix presents the pseudocode for the proposed equalisers. The complete source code will be made available following the acceptance of the associated journal paper. In what follows, we assume multichannel realisations, with layers following the architecture proposed in Chapter 4. However, the algorithms are applicable to single-channel equalisation as well, provided the linear and nonlinear layers are adequately defined. The IVSTF equaliser in Figure 6.3 (b) is implemented as follows:

Algorithm: IVSTF

Require: signal
  signal_fd = to_fd(signal)
  sig_out = total_cd_layer(signal_fd)
  for each step $m = 1 : L_{\text{sp}} * N_{\text{StpS}}$ **do**
    e = first_linear_layers[m](signal_fd)
    p = to_fd (nonlinear_layers[m](from_fd(e)))
    sig_out += last_linear_layers[m](p)
  return sig_out

Figure C.1: Pseudocode of the IVSTF algorithm [88].

Here, to_fd() and from_fd() are helper functions to apply FFT/iFFTs on the signal. The layer total_cd_layer() filters each channel i with the transfer function in Eq. 4.9 with the dispersion corresponding to a length NL_{sp}. The remaining multichannel layers are retrieved from lists. The lists first_linear_layers() and first_linear_layers() contain previously initialised frequency domain layers, while nonlinear_layers() contains the time-domain nonlinear layers. The NLC flag gives the option to enable or disable nonlinearity equalisation. The equalisers FE IVSTF and simIVSTF, depicted in Figure 6.4 (a) and (b), are implemented as follows:

In the simIVSTF algorithm, two separate for loops are employed: One of them computes the outputs of the nonlinear branches, while the other one removes the CD of each span and adds the nonlinear branch outputs. Two copies of the signal are created: y and y2, one for use in each for loop. We allow the possibility for the number of branches to be lower than the number of steps. This feature could be useful to reduce the complexity of

Algorithm: FE IVSTF

Require: signal
  signal_fd = to_fd(signal)
  sig_out = total_cd_layer(signal_fd)
  for each step $m = 1 : L_{\text{sp}} * N_{\text{StpS}}$ **do**
    e = first_linear_layers[m](signal_fd)
    f = first_td_linear_layers[m](from_fd(e))
    p = nonlinear_layers[m](f)
    g = to_fd(last_td_linear_layers[m](p))
    sig_out += last_linear_layers[m](g)
  return from_fd(sig_out)

Algorithm: simIVSTF

Require: signal
  y2 = y = signal
  branch = {}
  for each step $m = 1 : N_{\text{branches}}$ **do**
    y = branch_linear_layer[m](y)
    branch[m] = nonlinear_layers[m](y)
  for each step $n = 1 : L_{\text{sp}} * N_{\text{StpS}}$
    y2 = trunk_linear_layer[n](y2) +
    branch[n]
  return y2

Figure C.2: Pseudocode of the FE IVSTF and simIVSTF algorithms. FE IVSTF applies short FIR CD filters using convolutional layers before and after the nonlinear layer. The simIVSTF uses only convolutional layers, with separate loops for applying the filters associated with the linear and nonlinear branches.

the algorithm: the nonlinear contributions of the last fibre spans are expected to be less important and could be omitted.

Appendix D

Code Implementations

This appendix presents code implementations for several key operations employed in our MIMO equalisers. The custom layers employed in learned equalisers are implemented using the Tensorflow 2 framework as detailed in Sec. 4.2.3. The single channel layers depicted in Fig. 4.10 (a) apply delay compensation and CD filtering, as the following code shows:

```
class sc_linear_layer_td(tf.keras.layers.Layer):
    """
    This custom layer implements the linear step applied to a single channel in a MIMO equaliser.
    The layer applies circular shifting, a fractional delay filter, and a CD filter.
    """

    def __init__(self, cd_filter, freq, model_params, tx_params, layertype, step_number = 0, ch = 0):
        super(sc_linear_layer_td, self).__init__()

        beta2 = tx_params["beta2"]
        d_len = model_params["dlen"]
        fsampd = tx_params["fsamp_d"]

        # Compute the integer and fractional delay components in samples
        self.int_delay_samples, delay_remainder_samples = wo_delay_calculation(beta2, freq, fsampd, d_len)

        # Create a symmetric FIR CD filter
        h_real_right_half = tf_real_symmetric_filter(tf.math.real(cd_filter))
        h_imag_right_half = tf_real_symmetric_filter(tf.math.imag(cd_filter))

        cd_filter_cmplx = tf.stack([tf.cast(h_real_right_half, tf.float32), tf.cast(h_imag_right_half, tf.float32)],
                                   axis=1)

        self.cd_filter = tf.Variable(initial_value=cd_filter_cmplx, dtype=tf.float32, trainable=model_params["trainable_ln_steps"],
                                     name='sc_{a}_layer_step_{b}_ch_{c}'.format(a = layertype, b=step_number, c = ch))

        # Create a fractional delay windowed sinc filter
```

```
        fd_filter = fd_sinc_filter(model_params["fd_filter_length"],
        ↪ model_params["fd_bandwidth"], delay_remainder_samples)

        self.interp_filter = tf.Variable(initial_value=fd_filter,
        ↪ dtype=tf.float32,
        ↪ trainable=model_params["trainable_frac_delay_filter"])

    def call(self, signal):

        signal = tf.roll(signal, int(self.int_delay_samples), 1)  # addresses the
        ↪ integer delay

        if int(self.delay_samples) != 0:
            signal = cconv(signal, self.interp_filter)  # addresses the
            ↪ fractional delay if present

        sig_out = cconv(signal, self.cd_filter)  # addresses the chromatic
        ↪ dispersion

        return sig_out
```

By comparison, single-channel SPM and XPM layers (shown inside the multichannel nonlinear layer of Fig. 4.10 (b)) apply a single filter:

```
class sc_nonlinear_layer(tf.keras.layers.Layer):
    """
    A custom layer to apply a real-valued filter to a single-channel.
    """
    def __init__(self, nl_filter, trainable=True, step_number= 0, ch = 0,
    ↪ second_ch = 0):
        super(sc_nonlinear_layer, self).__init__()

        #Define the filter as a trainable variable
        self.nl_filter = tf.Variable(initial_value=nl_filter, dtype=tf.float32,
        ↪ trainable=trainable,
        ↪ name='sc_nl_layer_{a}_ch_{b}_to_ch_{c}'.format(a=step_number, b = ch,
        ↪ c = second_ch))

    def __call__(self, sig_pwr):

        filtered_pwr = cconv(sig_pwr, self.nl_filter)

        return filtered_pwr
```

We have explored how to efficiently implement MIMO schemes in Tensorflow. To avoid the nested for loops to traverse over channel pairs, we attempted to vectorise the calculation of XPM contributions. The signal power combinations required for each XPM contribution in the equaliser derived in Sec. 6.2 can be calculated as follows:

$$\sum_{q \neq m}^{N_{\text{ch}}} |U_q|^2 = \mathbf{M} \cdot \mathbf{P} \tag{D.1}$$

We define the mask as $\mathbf{M} = \mathbf{1} - \mathbf{I}$, where $\mathbf{1}$ is the all-ones matrix and $\mathbf{I}$ is the identity matrix. We note that the coefficients of this mask can be interpreted as the weight of each channel contribution. In future implementations, the weights of this mask may be learned along with the filter coefficients during the gradient-based optimisation process. The code is provided below.

```
def xpm_vectorized(sig_c):
    """
    Computes, in a vectorised manner, power combinations for the XPM phase shift.
    Parameters:
    sig_c: Signal tensor of shape (Nch, Nsamp)
    Returns:
    tf.Tensor: Tensor containing the total XPM phase shift for each channel
    """
    ones_array = np.ones((Nch, Nch)) # Array of shape (Nch, Nch) filled with ones
    id_array = np.identity(Nch) # Identity matrix for self-contributions
    mask = ones_array - id_array # Mask tensor of shape (Nch, Nch) with diagonal set to zero

    sig_power = tf.square(tf.abs(sig_c)) # Signal powers

    pwr_combinations = mask @ sig_power # XPM signal power combinations

    return pwr_combinations
```

Appendix E

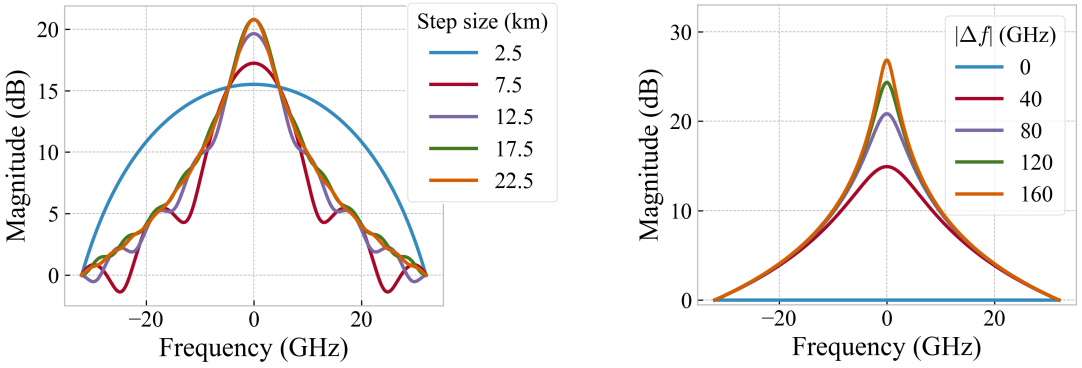
Response of Analytical XPM Filters

Analytical transfer functions for the filtering of XPM contributions can be obtained for MIMO DBP. The resulting nonlinear phase shift is given by [93]

$$\phi_m^{NL} = \mathcal{F}^{-1}\left[\sum \mathcal{F}\left(|E_q(t,z)|^2\right) W_{m,q}(\omega,h)\right] \tag{E.1}$$

$$W_{m,q}(\omega,h) = \begin{cases} \gamma h_{\text{eff}} & \text{for } q = m \\ 2\gamma \frac{e^{(\alpha+id_{m,q}\omega)h}-1}{\alpha+id_{m,q}\omega} & \text{for } q \neq m \end{cases} \tag{E.2}$$

Here, h is the step size. Figure E.1 shows the amplitude response of the transfer functions $W_{m,q}$ for different channel spacings and step sizes. The filters exhibit a low-pass response, symmetric around the center frequency. The shape of the response depends on the step size: smaller step sizes produce a Gaussian-like shape, while larger step sizes result in sharper cutoffs. Moreover, the filter magnitude increases with channel spacing, indicating that the filter amplifies contributions from more distant channels. Similar analytical filters could be derived for the multichannel IVSTF model in future efforts.



(a) (b)

Figure E.1: Magnitude response of the $W_{m,q}$ filters introduced in [93]

List of References

[1] E. Agrell, M. Karlsson, F. Poletti, S. Namiki, X. V. Chen, L. A. Rusch, B. Puttnam, P. Bayvel, L. Schmalen, Z. Tao, F. R. Kschischang, A. Alvarado, B. Mukherjee, R. Casellas, X. Zhou, D. v. Veen, G. Mohs, E. Wong, A. Mecozzi, M.-S. Alouini, E. Diamanti, and M. Uysal. Roadmap on optical communications. *J. Opt.*, 26(9): 093001, July 2024. ISSN 2040-8986. doi:10.1088/2040-8986/ad261f.

[2] A. Amari, P. Ciblat, and Y. Jaouën. Fifth-order Volterra series based nonlinear equalizer for long-haul high data rate optical fiber communications. In *2014 48th Asilomar Conference on Signals, Systems and Computers*, pages 1367–1371, Nov. 2014. doi:10.1109/ACSSC.2014.7094684. ISSN: 1058-6393.

[3] A. Bakhshali, W.-Y. Chan, J. C. Cartledge, M. O'Sullivan, C. Laperle, A. Borowiec, and K. Roberts. Frequency-domain Volterra-based equalization structures for efficient mitigation of intrachannel Kerr nonlinearities. *J. Lightwave Technol.*, 34(8):1770–1777, Apr 2016.

[4] A. Balatsoukas-Stimming and C. Studer. Deep unfolding for communications systems: A survey and some new directions. In *2019 IEEE International Workshop on Signal Processing Systems (SiPS)*, pages 266–271, 2019. doi:10.1109/SiPS47522.2019.9020494.

[5] P. Bayvel, R. Maher, T. Xu, G. Liga, N. A. Shevchenko, D. Lavery, A. Alvarado, and R. I. Killey. Maximizing the optical network capacity. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2062): 20140440, 2016. doi:10.1098/rsta.2014.0440.

[6] P. W. Berenguer, M. Nölle, L. Molle, T. Raman, A. Napoli, C. Schubert, and J. K. Fischer. Nonlinear digital pre-distortion of transmitter components. *J. Lightwave Technol.*, 34(8):1739–1745, Apr. 2016. ISSN 1558-2213. doi:10.1109/JLT.2015.2510962.

[7] J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, 13(10):281–305, 2012. URL `http://jmlr.org/papers/v13/bergstra12a.html`.

[8] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

[9] B. I. Bitachon, A. Ghazisaeidi, M. Eppenberger, B. Baeuerle, M. Ayata, and J. Leuthold. Deep learning based digital backpropagation demonstrating SNR gain at low complexity in a 1200 km transmission link. *Opt. Express*, 28(20):29318–29334, Sep 2020. doi:10.1364/OE.401667.

[10] D. Blalock, J. J. Gonzalez Ortiz, J. Frankle, and J. Guttag. What is the state of neural network pruning? In *Proceedings of Machine Learning and Systems*, volume 2, pages 129–146, 2020.

[11] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8), 2019. ISSN 2079-9292. doi:10.3390/electronics8080832.

[12] N. Castro and S. Sygletos. A novel learned Volterra-based scheme for time-domain nonlinear equalization. In *Conference on Lasers and Electro-Optics*, page SF3M.1, San Jose, California, 2022. Optica Publishing Group. ISBN 978-1-957171-05-0.

[13] N. Castro and S. Sygletos. Learned Volterra equalization for WDM systems. In *2023 Asia Communications and Photonics Conference/2023 International Photonics and Optoelectronics Meetings (ACP/POEM)*, pages 1–4, 2023.

[14] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Design aspects of frequency-domain learned MIMO Volterra equalisers. In *CLEO 2024*, page JTu2A.87. Optica Publishing Group, 2024.

[15] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Field-enhanced filtering in MIMO learned Volterra nonlinear equalisation of multi-wavelength systems. In *ECOC 2024; 50th European Conference on Optical Communication*, pages 902–905, 2024.

[16] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Design of time-domain learned Volterra equalisers for WDM systems. In *2024 International Conference on Optical Network Design and Modeling (ONDM)*, pages 1–3, 2024. doi:10.23919/ONDM61578.2024.10582691.

[17] N. Castro, S. Boscolo, A. D. Ellis, and S. Sygletos. Learned Volterra models for nonlinearity equalization in wavelength-division multiplexed systems. *Opt. Express*, 33(8):16717–16737, Apr 2025. doi:10.1364/OE.554077.

[18] S. Civelli, E. Forestieri, A. Lotsmanov, D. Razdoburdin, and M. Secondini. Multichannel digital backpropagation with XPM-aware ESSFM. In *2021 17th International Symposium on Wireless Communication Systems (ISWCS)*, pages 1–6, 2021. doi:10.1109/ISWCS49558.2021.9562209.

[19] S. Civelli, D. P. Jana, E. Forestieri, and M. Secondini. Coupled-band ESSFM for low-complexity DBP. In *50th European Conference on Optical Communications (ECOC 2024)*, 2024.

[20] S. Civelli, D. P. Jana, E. Forestieri, and M. Secondini. A new twist on low-complexity digital backpropagation. *Journal of Lightwave Technology*, pages 1–14, 2025. doi:10.1109/JLT.2025.3542166.

[21] D. de Arruda Mello and F. Barbosa. *Digital Coherent Optical Systems: Architecture and Algorithms*. Optical Networks. Springer International Publishing, 2021. ISBN 9783030665418.

[22] S. Deligiannidis, A. Bogris, C. Mesaritakis, and Y. Kopsinis. Compensation of fiber nonlinearities in digital coherent systems leveraging long short-term memory neural networks. *J. Lightwave Technol.*, 38(21):5991–5999, 2020. doi:10.1109/JLT.2020.3007919.

[23] S. Deligiannidis, C. Mesaritakis, and A. Bogris. Performance and complexity analysis of bi-directional recurrent neural network models versus Volterra nonlinear equalizers in digital coherent systems. *J. Lightwave Technol.*, 39(18):5791–5798, Sept. 2021. ISSN 1558-2213. doi:10.1109/JLT.2021.3092415.

[24] S. Deligiannidis, K. R. H. Bottrill, K. Sozos, C. Mesaritakis, P. Petropoulos, and A. Bogris. Multichannel nonlinear equalization in coherent WDM systems based on bi-directional recurrent neural networks. *J. Lightwave Technol.*, 42(2):541–549, Jan. 2024. ISSN 1558-2213. doi:10.1109/JLT.2023.3318559.

[25] L. B. Du and A. J. Lowery. Improved single channel backpropagation for intra-channel fiber nonlinearity compensation in long-haul optical communication systems. *Opt. Express*, 18(16):17075–17088, Aug 2010. doi:10.1364/OE.18.017075.

[26] L. B. Du, D. Rafique, A. Napoli, B. Spinnler, A. D. Ellis, M. Kuschnerov, and A. J. Lowery. Digital fiber nonlinearity compensation: Toward 1-Tb/s transport. *IEEE Signal Processing Magazine*, 31(2):46–56, Mar. 2014. ISSN 1558-0792. doi:10.1109/MSP.2013.2288110.

[27] H. Dzieciol, T. Koike-Akino, Y. Wang, and K. Parsons. Inverse regular perturbation with ML-assisted phasor correction for fiber nonlinearity compensation. *Opt. Lett.*, 47(14):3471–3474, Jul 2022. doi:10.1364/OL.460929.

[28] A. Eghbali, H. Johansson, O. Gustafsson, and S. J. Savory. Optimal least-squares FIR digital filters for compensation of chromatic dispersion in digital coherent optical receivers. *J. Lightwave Technol.*, 32(8):1449–1456, 2014. doi:10.1109/JLT.2014.2307916.

[29] A. Ellis, A. Ali, M. Tan, N. Salgado, and S. Sygletos. Mitigation of nonlinear effects in optical communications using digital and optical techniques. In *Optica Advanced Photonics Congress 2022*, page NeTu3D.4. Optica Publishing Group, 2022.

[30] A. D. Ellis, M. E. McCarthy, M. A. Z. A. Khateeb, M. Sorokina, and N. J. Doran. Performance limits in optical communications due to fiber nonlinearity. *Adv. Opt. Photon.*, 9(3):429–503, Sep 2017. doi:10.1364/AOP.9.000429.

[31] T. A. Eriksson, H. Bülow, and A. Leven. Applying neural networks in optical communication systems: Possible pitfalls. *IEEE Photonics Technology Letters*, 29(23): 2091–2094, Dec. 2017. ISSN 1941-0174. doi:10.1109/LPT.2017.2755663.

[32] H. Escalante, S. Escalera, I. Guyon, X. Baró, Y. Güçlütürk, U. Güçlü, and M. van Gerven. *Explainable and Interpretable Models in Computer Vision and Machine Learning*. The Springer Series on Challenges in Machine Learning. Springer International Publishing, 2018. ISBN 9783319981314.

[33] R.-J. Essiambre, G. Kramer, P. J. Winzer, G. J. Foschini, and B. Goebel. Capacity limits of optical fiber networks. *J. Lightwave Technol.*, 28(4):662–701, 2010. doi:10.1109/JLT.2009.2039464.

[34] Q. Fan. *Digital Signal Processing for Long-Haul Optical Communications Using Deep Learning*. PhD thesis, The Hong Kong Polytechnic University, 2022.

[35] Q. Fan, G. Zhou, T. Gui, C. Lu, and A. P. T. Lau. Advancing theoretical understanding and practical performance of signal processing for nonlinear optical communications through machine learning. *Nat. Commun.*, 11(1):3694, July 2020. ISSN 2041-1723. doi:10.1038/s41467-020-17516-7.

[36] Q. Fan, C. Lu, and A. P. T. Lau. Combined neural network and adaptive DSP training for long-haul optical communications. *J. Lightwave Technol.*, 39(22):7083–7091, 2021. doi:10.1109/JLT.2021.3111437.

[37] J. A. Fleck, J. R. Morris, and M. D. Feit. Time-dependent propagation of high energy laser beams through the atmosphere. *Appl. Phys.*, 10:129–160, 1976. doi:10.1007/BF00896333.

[38] C. Fougstedt, A. Sheikh, P. Johannisson, A. G. i Amat, and P. Larsson-Edefors. Power-efficient time-domain dispersion compensation using optimized FIR filter implementation. In *Advanced Photonics 2015*, page SpT3D.3. Optica Publishing Group, 2015. doi:10.1364/SPPCOM.2015.SpT3D.3.

[39] C. Fougstedt, M. Mazur, L. Svensson, H. Eliasson, M. Karlsson, and P. Larsson-Edefors. Time-domain digital back propagation: Algorithm and finite-precision implementation aspects. In *2017 Optical Fiber Communications Conference and Exhibition (OFC)*, pages 1–3, 2017.

[40] P. Freire, S. Srivallapanondh, B. Spinnler, A. Napoli, N. Costa, J. E. Prilepsky, and S. K. Turitsyn. Computational complexity optimization of neural network-based equalizers in digital signal processing: A comprehensive approach. *J. Lightwave Technol.*, 42(12):4177–4201, 2024. doi:10.1109/JLT.2024.3386886.

[41] P. J. Freire, V. Neskornuik, A. Napoli, B. Spinnler, N. Costa, G. Khanna, E. Riccardi, J. E. Prilepsky, and S. K. Turitsyn. Complex-valued neural network design for mitigation of signal distortions in optical links. *J. Lightwave Technol.*, 39(6):1696–1705, 2021. doi:10.1109/JLT.2020.3042414.

[42] 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. *J. Lightwave Technol.*, 39(19):6085–6096, 2021. doi:10.1109/JLT.2021.3096286.

[43] 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)*, pages 1–3, June 2021.

[44] 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 J. Sel. Top. Quantum Electron.*, 28:1–23, 2022. doi:10.1109/JSTQE.2022.3174268.

[45] P. J. Freire, J. E. Prilepsky, Y. Osadchuk, S. K. Turitsyn, and V. Aref. Deep neural network-aided soft-demapping in coherent optical systems: Regression versus classification. *IEEE Trans. Commun.*, 70(12):7973–7988, 2022. doi:10.1109/TCOMM.2022.3213284.

[46] P. J. Freire, A. Napoli, B. Spinnler, M. Anderson, D. A. Ron, 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. *J. Lightwave Technol.*, 41(14):4557–4581, 2023. doi:10.1109/JLT.2023.3234327.

[47] S. Fujisawa, F. Yaman, H. G. Batshon, M. Tanio, N. Ishii, C. Huang, T. Ferreira de Lima, Y. Inada, P. Prucnal, N. Kamiya, and T. Wang. Weight pruning techniques towards photonic implementation of nonlinear impairment compensation using neural networks. *J. Lightwave Technol.*, pages 1–1, 2021. ISSN 1558-2213. doi:10.1109/JLT.2021.3117609.

[48] K. Fukushima and S. Miyake. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and Cooperation in Neural Nets*, pages 267–285, Berlin, Heidelberg, 1982. Springer Berlin Heidelberg. ISBN 978-3-642-46466-9.

[49] L. Ge, W. Zhang, Y. Zhang, C. Liang, J. Du, and Z. He. Compressed nonlinear equalizers for optical interconnects: Efficiency and stability. In *2020 Optical Fiber Communications Conference and Exhibition (OFC)*, pages 1–3, Mar. 2020.

[50] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer. A survey of quantization methods for efficient neural network inference. *arXiv preprint arXiv:2103.13630*, 2021. URL `https://arxiv.org/abs/2103.13630`.

[51] E. Giacoumidis, J. Wei, I. Aldaya, and L. P. Barry. Exceeding the nonlinear Shannon-limit in coherent optical communications by MIMO machine learning. *arXiv preprint arXiv:1802.09120*, 2019. URL `https://arxiv.org/abs/1802.09120`.

[52] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In G. Gordon, D. Dunson, and M. Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 315–323, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR.

[53] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[54] F. P. Guiomar and A. N. Pinto. Simplified Volterra series nonlinear equalizer for polarization-multiplexed coherent optical systems. *J. Lightwave Technol.*, 31(23): 3879–3891, Dec. 2013. ISSN 1558-2213. doi:10.1109/JLT.2013.2288781.

[55] F. P. Guiomar, J. D. Reis, A. L. Teixeira, and A. N. Pinto. Mitigation of intra-channel nonlinearities using a frequency-domain Volterra series equalizer. *Opt. Express*, 20(2): 1360–1369, Jan 2012. doi:10.1364/OE.20.001360.

[56] F. P. Guiomar, S. B. Amado, C. S. Martins, and A. N. Pinto. Time-domain Volterra-based digital backpropagation for coherent optical systems. *J. Lightwave Technol.*, 33 (15):3170–3181, 2015. doi:10.1109/JLT.2015.2435520.

[57] C. Häger. LDBP. `https://github.com/chaeger/LDBP`, 2023.

[58] C. Häger and H. D. Pfister. Physics-based deep learning for fiber-optic communication systems. *IEEE Journal on Selected Areas in Communications*, 39(1):280–294, 2021. doi:10.1109/JSAC.2020.3036950.

[59] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. *arXiv preprint arXiv:1510.00149*, 2015. URL `https://arxiv.org/abs/1510.00149`.

[60] S. Haykin. Signal processing: where physics and mathematics meet. *IEEE Signal Processing Magazine*, 18(4):6–7, 2001. doi:10.1109/MSP.2001.939832.

[61] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu. Model-driven deep learning for physical layer communications. *IEEE Wireless Communications*, 26(5):77–83, 2019.

[62] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015. doi:10.1109/ICCV.2015.123.

[63] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, Jan. 1989. ISSN 08936080. doi:10.1016/0893-6080(89)90020-8.

[64] X. Huang, W. Jiang, X. Yi, J. Zhang, T. Jin, Q. Zhang, B. Xu, and K. Qiu. Design of fully interpretable neural networks for digital coherent demodulation. *Opt. Express*, 30(20):35526–35538, Sep 2022. doi:10.1364/OE.472406.

[65] C. Häger and H. D. Pfister. Deep learning of the nonlinear Schrödinger equation in fiber-optic communications. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 1590–1594, June 2018. doi:10.1109/ISIT.2018.8437734. ISSN: 2157-8117.

[66] C. Häger and H. D. Pfister. Nonlinear interference mitigation via deep neural networks. In *2018 Optical Fiber Communications Conference and Exposition (OFC)*, pages 1–3, Mar. 2018.

[67] C. Häger and H. D. Pfister. Wideband time-domain digital backpropagation via subband processing and deep learning. In *2018 European Conference on Optical Communication (ECOC)*, pages 1–3, Sept. 2018. doi:10.1109/ECOC.2018.8535251.

[68] M. Ibnkahla. Applications of neural networks to digital communications – a survey. *Signal Processing*, 80(7):1185–1215, 2000. ISSN 0165-1684. doi:10.1016/S0165-1684(00)00030-X.

[69] T. Inoue, R. Matsumoto, and S. Namiki. Learning-based digital back propagation to compensate for fiber nonlinearity considering self-phase and cross-phase modulation for wavelength-division multiplexed systems. *Opt. Express*, 30(9):14851–14872, Apr. 2022. ISSN 1094-4087. doi:10.1364/OE.454841. Publisher: Optica Publishing Group.

[70] E. Ip. Nonlinear compensation using backpropagation for polarization-multiplexed transmission. *J. Lightwave Technol.*, 28(6):939–951, Mar. 2010. ISSN 1558-2213. doi:10.1109/JLT.2010.2040135.

[71] E. Ip and J. M. Kahn. Digital equalization of chromatic dispersion and polarization mode dispersion. *J. Lightwave Technol.*, 25(8):2033–2043, 2007. doi:10.1109/JLT.2007.900889.

[72] E. Ip and J. M. Kahn. Compensation of dispersion and nonlinear impairments using digital backpropagation. *J. Lightwave Technol.*, 26(20):3416–3425, Oct. 2008. ISSN 1558-2213. doi:10.1109/JLT.2008.927791.

[73] E. Ip, N. Bai, and T. Wang. Complexity versus performance tradeoff for fiber nonlinearity compensation using frequency-shaped, multi-subband backpropagation. In *2011 Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference*, pages 1–3, 2011.

[74] S. Jansen, D. van den Borne, P. Krummrich, S. Spalter, G.-D. Khoe, and H. de Waardt. Long-haul DWDM transmission systems employing optical phase conjugation. *IEEE J. Sel. Top. Quantum Electron.*, 12(4):505–520, July 2006. ISSN 1558-4542. doi:10.1109/JSTQE.2006.876621.

[75] K. Kikuchi. Phase-diversity homodyne detection of multilevel optical modulation with digital carrier phase estimation. *IEEE J. Sel. Top. Quantum Electron.*, 12(4):563–570, 2006. doi:10.1109/JSTQE.2006.876307.

[76] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. URL `https://arxiv.org/abs/1412.6980`.

[77] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman. 1D convolutional neural networks and applications: A survey. *Mechanical Systems and Signal Processing*, 151:107398, 2021. ISSN 0888-3270. doi:10.1016/j.ymssp.2020.107398.

[78] P. Kootsookos and R. Williamson. FIR approximation of fractional sample delay systems. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 43(3):269–271, 1996. doi:10.1109/82.486473.

[79] R. Kudo, T. Kobayashi, K. Ishihara, Y. Takatori, A. Sano, and Y. Miyamoto. Coherent optical single carrier transmission using overlap frequency domain equalization for long-haul optical systems. *J. Lightwave Technol.*, 27(16):3721–3728, 2009. doi:10.1109/JLT.2009.2024091.

[80] M. Kuschnerov, M. Chouayakh, K. Piyawanno, B. Spinnler, E. de Man, P. Kainzmaier, M. S. Alfiad, A. Napoli, and B. Lankl. Data-aided versus blind single-carrier coherent receivers. *IEEE Photonics Journal*, 2(3):387–403, 2010. doi:10.1109/JPHOT.2010.2048308.

[81] T. Laakso, V. Valimaki, M. Karjalainen, and U. Laine. Splitting the unit delay [FIR/all pass filters design]. *IEEE Signal Processing Magazine*, 13(1):30–60, 1996. doi:10.1109/79.482137.

[82] Y. LeCun and Y. Bengio. *Convolutional networks for images, speech, and time series*, page 255–258. MIT Press, Cambridge, MA, USA, 1998. ISBN 0262511029.

[83] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989. doi:10.1162/neco.1989.1.4.541.

[84] S. Lennard, F. A. Barbosa, and F. Ferreira. Fully generalized machine learning-based equalization in coherent optical transmission. In *Advanced Photonics Congress 2024*, page SpM3G.2. Optica Publishing Group, 2024. doi:10.1364/SPPCOM.2024.SpM3G.2.

[85] X. Li, X. Chen, G. Goldfarb, E. Mateo, I. Kim, F. Yaman, and G. Li. Electronic post-compensation of WDM transmission impairments using coherent detection and digital signal processing. *Opt. Express*, 16(2):880–888, Jan 2008. doi:10.1364/OE.16.000880.

[86] G. Liga, T. Xu, A. Alvarado, R. I. Killey, and P. Bayvel. On the performance of multichannel digital backpropagation in high-capacity long-haul optical transmission. *Opt. Express*, 22(24):30053–30062, Dec 2014. doi:10.1364/OE.22.030053.

[87] X. Lin, S. Luo, S. K. O. Soman, O. A. Dobre, L. Lampe, D. Chang, and C. Li. Perturbation theory-aided learned digital back-propagation scheme for optical fiber nonlinearity compensation. *J. Lightwave Technol.*, 40(7):1981–1988, Apr. 2022. ISSN 0733-8724, 1558-2213. doi:10.1109/JLT.2021.3133475.

[88] L. Liu, L. Li, Y. Huang, K. Cui, Q. Xiong, F. N. Hauske, C. Xie, and Y. Cai. Intrachannel nonlinearity compensation by inverse Volterra series transfer function. *J. Lightwave Technol.*, 30(3):310–316, Feb. 2012. ISSN 1558-2213. doi:10.1109/JLT.2011.2182038.

[89] R. Maher, T. Xu, L. Galdino, M. Sato, A. Alvarado, K. Shi, S. J. Savory, B. C. Thomsen, R. I. Killey, and P. Bayvel. Spectrally shaped DP-16QAM super-channel transmission with multi-channel digital back-propagation. *Sci Rep*, 5(1):8214, July 2015. ISSN 2045-2322. doi:10.1038/srep08214.

[90] C. S. Martins, L. Bertignono, A. Nespola, A. Carena, F. P. Guiomar, and A. N. Pinto. Efficient time-domain DBP using random step-size and multi-band quantization. In *2018 Optical Fiber Communications Conference and Exposition (OFC)*, pages 1–3, 2018.

[91] E. Mateo, L. Zhu, and G. Li. Impact of XPM and FWM on the digital implementation of impairment compensation for WDM transmission using backward propagation. *Opt. Express*, 16(20):16124–16137, Sep 2008. doi:10.1364/OE.16.016124.

[92] E. F. Mateo and G. Li. Compensation of interchannel nonlinearities using enhanced coupled equations for digital backward propagation. *Appl. Opt.*, 48(25):F6, Sept. 2009. ISSN 0003-6935, 1539-4522. doi:10.1364/AO.48.0000F6.

[93] E. F. Mateo, F. Yaman, and G. Li. Efficient compensation of inter-channel nonlinear effects via digital backward propagation in WDM optical transmission. *Opt. Express*, 18(14):15144–15154, July 2010. ISSN 1094-4087. doi:10.1364/OE.18.015144. Publisher: Optica Publishing Group.

[94] E. F. Mateo, X. Zhou, and G. Li. Improved digital backward propagation for the compensation of inter-channel nonlinear effects in polarization-multiplexed WDM systems. *Opt. Express*, 19(2):570, Jan. 2011. ISSN 1094-4087. doi:10.1364/OE.19.000570.

[95] M. Matsumoto and T. Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Trans. Model. Comput. Simul.*, 8(1):3–30, Jan. 1998. ISSN 1049-3301. doi:10.1145/272991.272995.

[96] D. S. Millar, S. Makovejs, C. Behrens, S. Hellerbrand, R. I. Killey, P. Bayvel, and S. J. Savory. Mitigation of fiber nonlinearity using a digital coherent receiver. *IEEE J. Sel. Top. Quantum Electron.*, 16(5):1217–1226, 2010. doi:10.1109/JSTQE.2010.2047247.

[97] 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 Commun. Surv. Tutor.*, 21(2):1383–1408, 2019. doi:10.1109/COMST.2018.2880039.

[98] A. Napoli, Z. Maalej, V. A. J. M. 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. *J. Lightwave Technol.*, 32(7):1351–1362, 2014. doi:10.1109/JLT.2014.2301492.

[99] T. T. Nguyen, T. Zhang, E. Giacoumidis, A. A. Ali, M. Tan, P. Harper, L. P. Barry, and A. D. Ellis. Coupled transceiver-fiber nonlinearity compensation based on machine learning for probabilistic shaping system. *J. Lightwave Technol.*, 39(2):388–399, Jan. 2021. ISSN 1558-2213. doi:10.1109/JLT.2020.3029336.

[100] Z. Niu, H. Yang, L. Li, M. Shi, G. Xu, W. Hu, and L. Yi. Learnable digital signal processing: a new benchmark of linearity compensation for optical fiber communications. *Light: Science & Applications*, 13(1):188, Aug. 2024. ISSN 2047-7538. doi:10.1038/s41377-024-01556-5.

[101] A. V. Oppenheim and R. W. Schafer. *Digital Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1975.

[102] Z. Pan, B. Châtelain, M. Chagnon, and D. V. Plant. Volterra filtering for nonlinearity impairment mitigation in DP-16QAM and DP-QPSK fiber optic communication systems. In *2011 Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference*, pages 1–3, 2011.

[103] T. Parks and C. Burrus. *Digital Filter Design*. Topics in Digital Signal Processing. Wiley, 1987. ISBN 9780471828969.

[104] K. V. Peddanarappagari and M. Brandt-Pearce. Volterra series transfer function of single-mode fibers. *J. Lightwave Technol.*, 15(12):2232–2241, Dec. 1997. ISSN 1558-2213. doi:10.1109/50.643545.

[105] P. Poggiolini. The GN model of non-linear propagation in uncompensated coherent optical systems. *J. Lightwave Technol.*, 30(24):3857–3879, 2012. doi:10.1109/JLT.2012.2217729.

[106] J. Proakis and D. Manolakis. *Digital Signal Processing, Pearson New International Edition*. Pearson, 2013. ISBN 9781292025735.

[107] D. Rafique. Fiber nonlinearity compensation: Commercial applications and complexity analysis. *J. Lightwave Technol.*, 34(2):544–553, 2016. doi:10.1109/JLT.2015.2461512.

[108] D. Rafique, M. Mussolin, M. Forzati, J. Mårtensson, M. N. Chugtai, and A. D. Ellis. Compensation of intra-channel nonlinear fibre impairments using simplified

digital back-propagation algorithm. *Opt. Express*, 19(10):9453–9460, May 2011. doi:10.1364/OE.19.009453.

[109] A. Redyuk, E. Averyanov, O. Sidelnikov, M. Fedoruk, and S. Turitsyn. Compensation of nonlinear impairments using inverse perturbation theory with reduced complexity. *J. Lightwave Technol.*, 38(6):1250–1257, 2020. doi:10.1109/JLT.2020.2971768.

[110] J. D. Reis and A. L. Teixeira. Unveiling nonlinear effects in dense coherent optical WDM systems with Volterra series. *Opt. Express*, 18(8):8660–8670, Apr 2010. doi:10.1364/OE.18.008660.

[111] R. Robey and Y. Zamora. *Parallel and High Performance Computing*. Manning, 2021. ISBN 9781617296468.

[112] D. A. Ron, P. J. Freire, J. E. Prilepsky, M. Kamalian-Kopae, A. Napoli, and S. K. Turitsyn. Experimental implementation of a neural network optical channel equalizer in restricted hardware using pruning and quantization. *Sci Rep*, 12(1):8713, May 2022. ISSN 2045-2322. doi:10.1038/s41598-022-12563-0. Publisher: Nature Publishing Group.

[113] S. J. Savory. Digital filters for coherent optical receivers. *Opt. Express*, 16(2):804–817, Jan 2008. doi:10.1364/OE.16.000804.

[114] S. J. Savory. Digital coherent optical receivers: Algorithms and subsystems. *IEEE J. Sel. Top. Quantum Electron.*, 16(5):1164–1179, Sept. 2010. ISSN 1558-4542. doi:10.1109/JSTQE.2010.2044751.

[115] M. Schetzen. *The Volterra and Wiener Theories of Nonlinear Systems*. R.E. Krieger Publishing Company, 1989. ISBN 9780894643569.

[116] M. Secondini, D. Marsella, and E. Forestieri. Enhanced split-step Fourier method for digital backpropagation. In *ECOC 2014; European Conference on Optical Communication*, pages 1–3, 2014. doi:10.1109/ECOC.2014.6964122.

[117] M. Secondini, S. Rommel, G. Meloni, F. Fresi, E. Forestieri, and L. Potì. Single-step digital backpropagation for nonlinearity mitigation. *Photon Netw Commun*, 31(3): 493–502, June 2016. ISSN 1572-8188. doi:10.1007/s11107-015-0586-z.

[118] E. Sedov. *Machine Learning For Performance Improvement of Long-Haul End-to-End Optical Transmission Systems*. PhD thesis, Aston University, 2023.

[119] R. A. Shafik, M. S. Rahman, and A. R. Islam. On the extended relationships among EVM, BER and SNR as performance metrics. In *2006 International Conference on Electrical and Computer Engineering (ICECE)*, pages 408–411. IEEE, 2006. doi:10.1109/ICECE.2006.355657.

[120] A. Shahkarami, M. Yousefi, and Y. Jaouën. Complexity reduction over bi-RNN-based nonlinearity mitigation in dual-pol fiber-optic communications via a CRNN-based approach. *Optical Fiber Technology*, 74:103072, 2022. ISSN 1068-5200. doi:https://doi.org/10.1016/j.yofte.2022.103072.

[121] A. Sheikh, C. Fougstedt, A. G. i. Amat, P. Johannisson, P. Larsson-Edefors, and M. Karlsson. Dispersion compensation FIR filter with improved robustness to coefficient quantization errors. *J. Lightwave Technol.*, 34(22):5110–5117, 2016. doi:10.1109/JLT.2016.2599276.

[122] G. Shulkind and M. Nazarathy. Estimating the Volterra series transfer function over coherent optical OFDM for efficient monitoring of the fiber channel nonlinearity. *Opt. Express*, 20(27):29035–29062, Dec 2012. doi:10.1364/OE.20.029035.

[123] O. Sidelnikov, A. Redyuk, and S. Sygletos. Equalization performance and complexity analysis of dynamic deep neural networks in long haul transmission systems. *Opt. Express*, 26(25):32765–32776, Dec. 2018. ISSN 1094-4087. doi:10.1364/OE.26.032765. Publisher: Optical Society of America.

[124] O. Sidelnikov, A. Redyuk, S. Sygletos, M. Fedoruk, and S. Turitsyn. Advanced convolutional neural networks for nonlinearity mitigation in long-haul WDM transmission systems. *J. Lightwave Technol.*, 39(8):2397–2406, Apr. 2021. ISSN 1558-2213. doi:10.1109/JLT.2021.3051609.

[125] E. Sillekens, W. Yi, D. Semrau, A. Ottino, B. Karanov, D. Lavery, L. Galdino, P. Bayvel, R. I. Killey, S. Zhou, K. Law, and J. Chen. Time-domain learned digital back-propagation. In *2020 IEEE Workshop on Signal Processing Systems (SiPS)*, pages 1–4, Oct. 2020. doi:10.1109/SiPS50750.2020.9195253. ISSN: 2374-7390.

[126] M. Sorokina, S. Sygletos, and S. Turitsyn. Sparse identification for nonlinear optical communication systems: SINO method. *Opt. Express*, 24(26):30433, Dec. 2016. ISSN 1094-4087. doi:10.1364/OE.24.030433.

[127] B. Spinnler. Equalizer design and complexity for digital coherent receivers. *IEEE Journal of Selected Topics in Quantum Electronics*, 16(5):1180–1192, 2010. doi:10.1109/JSTQE.2009.2035931.

[128] S. Srivallapanondh, P. Freire, B. Spinnler, N. Costa, W. Schairer, A. Napoli, S. K. Turitsyn, and J. E. Prilepsky. Experimental validation of XPM mitigation using a generalizable multi-task learning neural network. *Opt. Lett.*, 49(24):6900–6903, Dec 2024. doi:10.1364/OL.535396.

[129] S. Srivallapanondh, P. J. Freire, B. Spinnler, N. Costa, A. Napoli, S. K. Turitsyn, and J. E. Prilepsky. Parallelization of recurrent neural network-based equalizer for coherent optical systems via knowledge distillation. *J. Lightwave Technol.*, 42(7): 2275–2284, 2024. doi:10.1109/JLT.2023.3337604.

[130] N. Stojanovic, F. Karinou, Z. Qiang, and C. Prodaniuc. Volterra and Wiener equalizers for short-reach 100G PAM-4 applications. *J. Lightwave Technol.*, 35(21):4583–4594, Nov. 2017. ISSN 1558-2213. doi:10.1109/JLT.2017.2752363.

[131] Z. Tao, Y. Fan, X. Su, K. Zhang, C. Yang, T. Ye, J. Li, H. Nakashima, and T. Hoshida. Characterization, measurement and specification of device imperfections in optical coherent transceivers. *J. Lightwave Technol.*, 40(10):3163–3172, 2022. doi:10.1109/JLT.2022.3155454.

[132] TensorFlow. Writing a training loop from scratch, 2023. URL `https://www.tensorflow.org/guide/keras/writing_a_training_loop_from_scratch`. Accessed: 2024-10-29.

[133] M. Torbatian, D. Lavery, M. Osman, D. Yao, D. S. Millar, Y. Gao, A. Kakkar, Z. A. El-Sahn, C. Doggart, A. E. Morra, N. Abughalieh, S. Yang, X. Chen, R. Maher, H. Sun, K.-T. Wu, and P. Kandappan. Performance oriented DSP for flexible long haul coherent transmission. *J. Lightwave Technol.*, 40(5):1256–1272, 2022. doi:10.1109/JLT.2021.3134155.

[134] J. Tsimbinos and K. Lever. The computational complexity of nonlinear compensators based on the Volterra inverse. In *Proceedings of 8th Workshop on Statistical Signal and Array Processing*, pages 387–390, 1996. doi:10.1109/SSAP.1996.534897.

[135] R. Turner, D. Eriksson, M. McCourt, J. Kiili, E. Laaksonen, Z. Xu, and I. Guyon. Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020. In *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*, volume 133 of *Proceedings of Machine Learning Research*, pages 3–26. PMLR, 06–12 Dec 2021.

[136] A. Vannucci, P. Serena, and A. Bononi. The RP method: a new tool for the iterative solution of the nonlinear Schrödinger equation. *J. Lightwave Technol.*, 20(7):1102–1112, July 2002. ISSN 1558-2213. doi:10.1109/JLT.2002.800376.

[137] V. Vgenopoulou, M. S. Erkilinc, R. I. Killey, Y. Jaouen, I. Roudas, and I. Tomkos. Comparison of multi-channel nonlinear equalization using inverse Volterra series versus digital backpropagation in 400 gb/s coherent superchannel. In *ECOC 2016; 42nd European Conference on Optical Communication*, pages 1–3, 2016.

[138] V. Vgenopoulou, N. P. Diamantopoulos, I. Roudas, and S. Sygletos. MIMO nonlinear equalizer based on inverse Volterra series transfer function for coherent SDM systems. In *2019 Optical Fiber Communications Conference and Exhibition (OFC)*, pages 1–3, 2019.

[139] P. J. Winzer, D. T. Neilson, and A. R. Chraplyvy. Fiber-optic transmission and networking: the previous 20 and the next 20 years [Invited]. *Opt. Express*, 26(18):24190–24239, Sep 2018. doi:10.1364/OE.26.024190.

[140] C. Xie. Impact of nonlinear and polarization effects in coherent systems. *Opt. Express*, 19(26):B915–B930, Dec 2011. doi:10.1364/OE.19.00B915.

[141] T. Xu, G. Jacobsen, S. Popov, J. Li, E. Vanin, K. Wang, A. T. Friberg, and Y. Zhang. Chromatic dispersion compensation in coherent transmission system using digital filters. *Opt. Express*, 18(15):16243–16257, Jul 2010. doi:10.1364/OE.18.016243.

[142] T. Xu, G. Jacobsen, S. Popov, M. Forzati, J. Mårtensson, M. Mussolin, J. Li, K. Wang, Y. Zhang, and A. T. Friberg. Frequency-domain chromatic dispersion equalization using overlap-add methods in coherent optical system. *J. Opt. Commun.*, 32(2):131–135, 2011. doi:doi:10.1515/joc.2011.022.

[143] Z. Xu and J. Sun. Model-driven deep-learning. *Natl. Sci. Rev.*, 5(1):22–24, 08 2017. ISSN 2095-5138. doi:10.1093/nsr/nwx099.

[144] S. Zhang, F. Yaman, K. Nakamura, T. Inoue, V. Kamalov, L. Jovanovski, V. Vusirikala, E. Mateo, Y. Inada, and T. Wang. Field and lab experimental demonstration of nonlinear impairment compensation using neural networks. *Nat. Commun.*, 10(1):1–8, July 2019. ISSN 2041-1723. doi:10.1038/s41467-019-10911-9. Number: 1 Publisher: Nature Publishing Group.

[145] L. Zhu, X. Li, E. Mateo, and G. Li. Complementary FIR filter pair for distributed impairment compensation of WDM fiber transmission. *IEEE Photonics Technology Letters*, 21(5):292–294, 2009. doi:10.1109/LPT.2008.2010871.