Adaptive Compensation of Nonlinear Impairments in Fiber-Optic Systems

by

Hadi NekoeiQachkanloo

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2020

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Optical communication systems are vital for high rate telecommunication. Fiber-optic communication system is an excellent choice due to its low loss, high bandwidth, and robustness to electromagnetic interference. However, fiber-optic links suffer from linear and nonlinear impairments which limit their performance. Digital signal processing techniques can be used for linear impairments compensation. On the other hand, nonlinear impairment is much harder to tackle.

There exist two main nonlinear noise which is caused by Kerr effect. Each channel in the fiber-optic link has two poles namely Xpole and Ypole. In a single channel case, transmitted signal over each pole generates intensity-dependent noise on both poles which is called Self Phase Modulation (SPM) noise. On the other hand, when multiple signal channels co-propagate in a single fiber, the power fluctuations of one signal channel cause a phase shift to another channel, which is due to the Cross Phase Modulation (XPM) effect.

Through this thesis, our main contributions are as follows. Firstly, we utilize Low-density parity-check (LDPC) Coded Modulation with Iterative Damping and Decoding at receiver to overcome the nonlinear noise without any need for feedback to transmitter. In other words, after extracting the short-term mean of SPM noise, we modify the decoding system to accept a priori information which helps us to remove nonlinearity using demapping. In addition, we propose a joint detection method to compensate for SPM noise. In this method, we exploit two main statistical characteristics of noise samples which are space domain and time domain correlations to improve naive minimum distance detection. In the last chapter, we introduce an algorithm for learning an adaptive model of fiber which can help us not only improve the performance of pre-compensation but also reduce the complexity of the state-of-the-art method.

Acknowledgements

Firstly, I would like to express my sincere gratitude to my advisor Prof. Amir Khandani for the continuous support of my M.Sc study and related research, for his patience, motivation, and immense knowledge. I would like to acknowledge Ciena Corporation for providing a unique opportunity for me to do research side by side with their scientists as an internship program, their technical input, and provision of data used in this research. Especially, I would like to thank Maurice, Masoud, Mike, and Shahab for their valuable contribution to the last chapter of this work and their mentorship. Also I want to Thank Ali, Nima and Alireza for fruitful conversations, and their help. Last but not the least, I would like to thank my family for their never ending support and love.

This work has been financially supported through a joint investment by Ciena Corporation and Natural Sciences and Engineering Research Council of Canada (NSERC).

Dedication

This is dedicated to my mother and father, Mehrnesa and AbbasAli, for their unconditional love.

Table of Contents

List of Figures vii

1 Overview and Literature review 1

1.1 Overview . . . 1

2 Extracting Statistical Characteristics of Nonlinear Noise 5

2.1 Statistical Characteristics of SPM Noise . . . 6

2.1.1 First Order Characteristics . . . 6

2.2 Statistical characteristics of XPM noise . . . 7

2.2.1 Space domain correlation . . . 9

2.2.2 Time Domain Correlation . . . 9

2.2.3 Correlation between X and Y poles . . . 10

2.3 Detection using both space and time domain correlation . . . 10

2.4 Results . . . 10

2.5 Summary . . . 11

2.5.1 Conclusions . . . 12

2.5.2 List of contributions . . . 12

2.5.3 Future Research . . . 12

3 Iterative Decoding **15**

3.1 Introduction . . . 15

3.2 System Model . . . 16

3.2.1 Encoding structure . . . 16

3.2.2 Modulation . . . 18

3.2.3 Decoding . . . 18

3.3 Short-Mean Updating . . . 20

3.4 Results . . . 22

3.5 Summary and Conclusions . . . 22

3.5.1 List of contributions . . . 24

4 Learning an Adaptive Model of Fiber **25**

4.1 Introduction . . . 25

4.2 Model . . . 26

4.2.1 2D C-matrix to 3D C-matrix . . . 28

4.2.2 Complexity Reduction . . . 29

4.3 Results . . . 32

4.3.1 Iterative Pre-Compensation . . . 34

4.3.2 Hyper-parameter optimization . . . 35

4.4 Summary . . . 37

4.4.1 Conclusions . . . 38

4.4.2 List of contributions . . . 39

4.4.3 Future research . . . 39

References **40**

List of Figures

2.1 SPM noise added to a 16QAM constellation points after passing through a 5 span fiber . . . 6
2.2 Short-term mean . . . 8
2.3 XPM noise added to a 16QAM constellation points after passing through an ELEAF 25 span fiber . . . 9
2.4 The effectiveness of exploiting time and space domain correlation . . . 11
2.5 SPM noise distribution . . . 13
2.6 XPM noise distribution . . . 14

3.1 System model . . . 17
3.2 Symbol set partitioning . . . 19
3.3 Updating short-term means . . . 21
3.4 Experiments on 16QAM . . . 23
3.5 Experiments on 64QAM . . . 23

4.1 (a) Standard (b) Pre-compensation using Analytic C-matrix [1] (c) Our Algorithm . . . 27
4.2 Learned C-matrix for $l = 0, \pm 1$ . . . 29
4.3 Efficient learning of C-matrix . . . 30
4.4 Learned vs Analytic C-matrix . . . 31
4.5 Efficient learning of C-matrix . . . 32
4.6 (a) Standard (b) unrolled structure . . . 33

4.7 Deimos dual polarization 16QAM, eleaf 20 spans, pCt = 0.5, Truncation Length = 55 35
4.8 Deimos dual polarization 16QAM, ndsf 20 spans, pCt = 0.5, Truncation Length = 55 36
4.9 SPM estimation 37
4.10 Deimos dual polarization 16QAM, eleaf 20 spans, pCt = 0.5, Truncation Length = 55 38
11 Convergence of the iterative pre-compensation 45

1

Overview and Literature review

1.1 Overview

Optic communication systems are vital for high-rate telecommunication. Fiber-optic communication system is an excellent choice due to its low loss, high bandwidth, and robustness to the electromagnetic interference. However, fiber-optic links suffer from linear and nonlinear impairments which limit their performance. Digital signal processing techniques can be used for the linear impairments compensation. On the other hand, nonlinear impairments are much harder to tackle. At low signal power, transmission performance is limited by amplified spontaneous emission (ASE) noise, so the capacity can be enhanced by increasing signal power. At higher signal to noise ratios, fiber nonlinear effects are dominant and enhancing transmission performance becomes impossible by simply increasing signal power.

There exist two main nonlinear noise which is caused by Kerr effect. Kerr effect is the dependence of the refractive index on the intensity of the optical pulse. Each channel in the fiber-optic link has two poles namely X-pole and Y-pole. For a single channel case, transmitted signal over each pole generates intensity-dependent noise on the other pole which is called Self Phase Modulation (SPM) noise. On the other hand, when multiple signal channels co-propagate in a single fiber, the power fluctuations of one signal channel cause a phase shift to other channel, which is due to the Cross Phase Modulation (XPM) effect [2], and [3]. Added directly to the phase of a signal, nonlinear phase noise becomes a major limitation for phase-modulated optical communications [4], [5], [6].

On a fundamental level, the origin of nonlinear response is related to the anharmonic motion of bound electrons under the influence of the applied field. As a result, the total

polarization P induced by electric dipoles is nonlinear in the electric field E, but satisfies the more general relation ([7], [8])

$$P = \epsilon_o(\chi^1.E + \chi^2.EE + \chi^3.EEE + ...)$$

where ϵ_o is the vacuum permittivity and $\chi(j)_{j=1,2,...}$ is jth order susceptibility. The lowest-order nonlinear effects in optical fibers originate from the third-order susceptibility χ^3, which is responsible for phenomena such as third harmonic generation, four-wave mixing, and nonlinear refraction. Therefore, most of the nonlinear effects in optical fibers originate from nonlinear refraction, a phenomenon referring to the intensity dependence of the refractive index. The nonlinear phase shift is given by [8]

$$\phi_{NL} = n_2 k_0 L(|E_1|^2 + 2|E_2|^2)$$

which consists of two components resulted from SPM and XPM noise, respectively. Therefore, the fiber-optic links are influence by XPM noise two times more than SPM noise.

Various digital compensation techniques have been proposed in [9] to compensate for SPM. For XPM compensation, one of the common methods is using intensity-dependent phase-modulation, in which the intensity of the phase modulator is controlled by the received signals from other channels [10]. Moreover, for mitigating XPM some previous studies has used non-zero dispersion fiber to induce walk off [9]. Back propagation, a procedure introduced in [11] and [12] is another method to tackle XPM noise. This method uses inverse Schrodinger equation to estimate what signal has been transmitted [13]. This method compensates both linear and nonlinear impairments [12].

Iterative decoding procedure is based on "belief propagation" which consists of two decoders: inner decoder and outer decoder. Each of these decoders compute a posteriori probability (APP) of the information symbols or, more generally, a reliability value for each information symbol. The sequence of reliability values generated by the decoder is passed to the other one [14] as a suggestion . To improve the correctness of its decisions, each decoder has to be fed with information which does not originate from itself [15]. The concept of extrinsic information was introduced by [16] and [15] to identify the component of the generated reliability value which depends on redundant information introduced by the considered constituent code. The logarithmic likelihood ratio (LLR) as a natural reliability value may be exactly computed employing the BCJR algorithm [17].

As noted in [18], [19], and [20], modifying the soft demapping to accept a priori information can reduce the bit error rate. Nikopour et al. [21] shows that when the nonlinear characteristic of the transmitter is known, the nonlinear distortion is a deterministic function of the transmitted data. Therefore, by using the estimated data, after some initial

turbo decoding iterations, the turbo decoder partially compensates for the nonlinear noise. Inspired by these studies, we modify the soft demapping to compensate for nonlinear noise in fiber-optic channel.

Through this thesis, SPM and XPM noise compensation is explored by different DSP and Channel Coding techniques. This thesis is organized into three main chapters. In chapter 2, we propose a joint detection method to compensate for XPM noise. Section 2.1 and 2.2 describe main statistical characteristics of SPM and XPM samples which are space domain and time domain correlations. We exploit these characteristics in order to improve naive minimum distance detection and we report the experimental results in section 2.4. Our findings show that the statistical approach does not help significantly in desired signal to noise ratio. Also, we investigate higher order characteristics of SPM noise distribution by estimating its probability distribution more accurately which can be exploited in future research to improve the bit error rate.

Chapter 3 focuses on using LDPC-Coded modulation with Iterative Damping and Decoding to improve bit probabilities. Section 3.2 describes the system model including encoding structure, signal model and decoding structure. In section 3.3, we explain how to embed updating short-mean characteristic of SPM noise in iterative algorithm by adding a new block to the iterative decoder. Finally, we report the experimental results in section 3.4 which prove the effectiveness of our proposed algorithm. We utilize this form of channel coding to compensate for nonlinear noise in the fiber-optic links for two main reasons. Firstly, iterative decoding at the receiver obviates the need of feedback to the transmitter to pre-compensate for the SPM noise [9]. Secondly, the decoding system can be modified to accept a priori information which helps us to remove nonlinear noise using demapping.

Chapter 4 is dedicated to introducing a pre-compensation algorithm which instead of calculating C-matrix analytically, learns C-matrix based on the inputs and outputs of the channel. This makes the algorithm adaptive to different situations of the fiber. Then, we apply same complexity reduction methods as [1] to have a fair comparison in terms of performance and complexity. The other advantage of this method, which is explained in section 4.2.1, would be its flexibility of learning higher order parameters of the model of fiber which analytic C-matrix is not capable of.

Our main contributions include

- Extracting statistical characteristics of SPM and XPM noise.
- Modifying the iterative decoding algorithm to accept short-term mean characteristic of nonlinear noise as a priori

- Proposing an adaptive model of fiber for pre-compensation of nonlinear noise

2

Extracting Statistical Characteristics of Nonlinear Noise

Is it the fault of wine if a fool drinks it and goes stumbling into darkness?

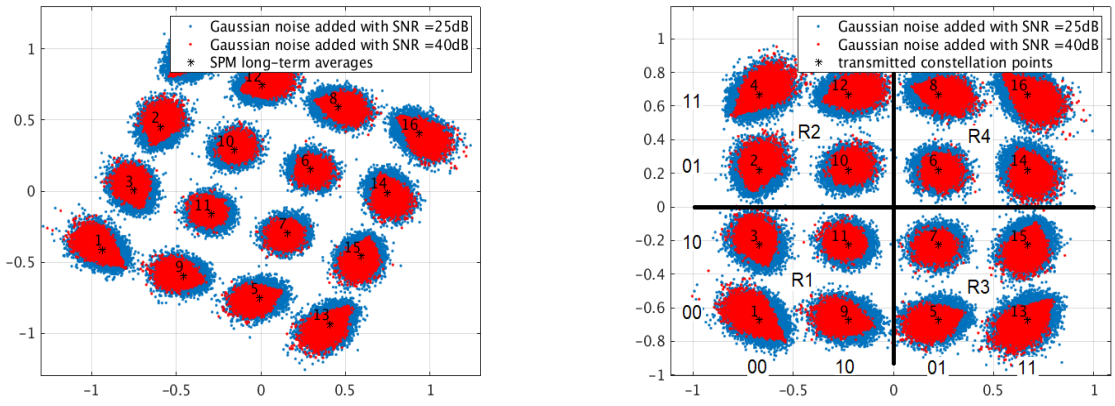
– Ibn Sina (Avicenna)

The most commonly used algorithm to decode transmitted signals over a linear channel is minimum distance detection. However, in the case of optical channels, minimum distance detection is not the best choice because of the nonlinear noise impairments. In this chapter, we show that the detection performance improves by considering the statistical features of SPM and XPM noise.

The non-linearity in a fibre optic links can be modeled using split-step model described in [22] and [9]. Assume that two streams of data symbols are launched on Xploe and Y-pole of two fiber channels: $\{A_x, A_y\}$ on channel A and $\{B_x, B_y\}$ on channel B. The equation for the non-linearity on the X-pole in the channel A at time zero has 6 components as follows.

$$\delta A_x(t) = SPM_x(t) + SPM_y(t) + \sum_w (XPM_{1w}(t) + XPM_{2w}(t) + XPM_{3w}(t) + XPM_{4w}(t)) \tag{2.1}$$

where $SPM_x(t)$ and $SPM_y(t)$ corresponds to the SPM noise at time t added by X-pole on itself and Y-pole on X-pole, respectively. The next 4 components are the results of XPM noise summed over all neighbours w.



(a) Received data points

(b) After subtracting the long-term mean

Figure 2.1: SPM noise added to a 16QAM constellation points after passing through a 5 span fiber

2.1 Statistical Characteristics of SPM Noise

SPM components in the non-linearity equation 2.1 can be computed as:

$$SPM_x(t) = \sum_{m,n} C^{spm}_{m,n} A_x(t+m) A_x(t+n) A^*_x(t+m+n) \tag{2.2}$$

$$SPM_y(t) = \sum_{m,n} C^{spm}_{m,n} A_x(t+m) A_y(t+n) A^*_y(t+m+n) \tag{2.3}$$

where A_x and A_y are the data symbols transmitted on X-pole and Y-pole of the base channel. In order to predict the noise behavior, one needs to investigate its statistical characteristics.

2.1.1 First Order Characteristics

Long-term mean: long-term mean is a primal characteristic for SPM as a random variable. It is obvious in the Figure 2.1a that the long-run means depends on constellation

points. The long-term mean is easily removable. Thus we do not consider it in our experiments. From this point, it is assumed that long-term mean is always deducted from the SPM.

Following calculations show that $\mathbb{E}[SPM]$ is dependant on the transmitted symbol or equivalently, the energy of transmitted symbol. Note that $\mathbb{E}[A_x(n)A_x^*(m+n)]$ is nonzero only if $m=0$ as the transmitted symbols are independent of each other due to the interleaving.

$$\begin{aligned}
\mathbb{E}[SPM(t)] &= \sum_{m=0,n} C^{spm}_{m=0,n} A_x(t)\,\mathbb{E}[A_x(t+n)A_x^*(t+n)] \\
&\quad + \sum_{m=0,n} C^{spm}_{m=0,n} A_x(t)\,\mathbb{E}[A_y(t+n)A_y^*(t+n)] \\
&\quad + \sum_{m,n=0} C^{spm}_{m,n=0} A_x(t)\,\mathbb{E}[A_x(t+m)A_x^*(t+m)] \\
&= A_x(t) \sum_{m=0,n} C^{spm}_{m=0,n} \big(\mathbb{E}[|A_x(t+n)|^2] + \mathbb{E}[|A_y(t+n)|^2]\big) \\
&\quad + A_x(t) \sum_{m,n=0} C^{spm}_{m,n=0} \big(\mathbb{E}[|A_x(t+m)|^2]\big)
\end{aligned}$$

Short-term mean: Sliding a window with length three over transmitted symbols, provides $16^3 = 4096$ different combination of transmitted symbols $\{A_x(t-1), A_x(t), A_x(t+1)\}$. For the sake of visualization, Assume that each of the transmitted symbols at time $t-1$ and $t+1$ are divided into four groups based on their region in Fig 2.1b. Therefore for a fixed $A_x(t)$ (e.g. $A_x(t) = \{-0.67, -0.67\}$), there will be $4^2 = 16$ different combinations of (R_i, R_j). As it can be seen in Fig 2.2, these 16 pairs of regions can be clustered to 4 main groups, separated with colors, which have different averages. As a result, the average of SPM with fixed $A_x(t)$ is dependant on $A_x(t-1)$ and $A_x(t+1)$ which can be used to compensate for the non-linearity noise.

2.2 Statistical characteristics of XPM noise

XPM components in the non-linearity equation 2.1 can be computed as:

$$XPM_{1w} = \sum_{m,n} C^{xpm_w}_{m,n} A_x(m) B_x(n) B_x^*(m+n) \tag{2.4}$$

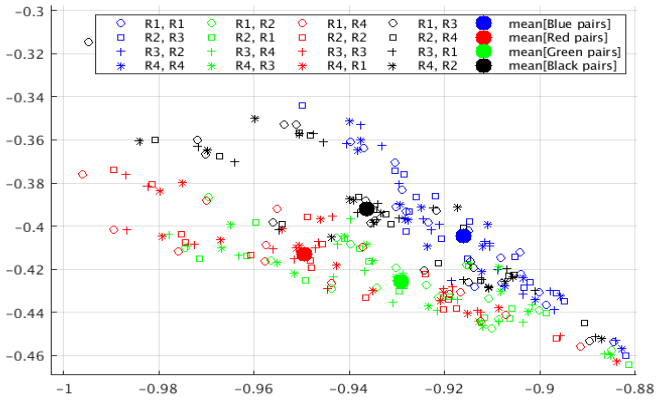


Figure 2.2: Short-term mean

$$XPM_{2w} = \sum_{m,n} C_{m,n}^{xpm_w} A_x(m) B_y(n) B_y^*(m+n) \tag{2.5}$$

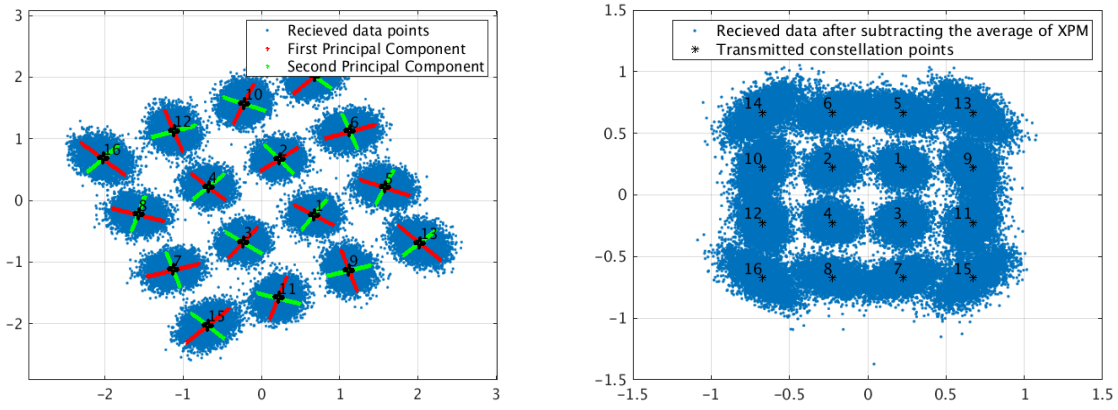
$$XPM_{3w} = \sum_{m,n} C_{m,n}^{xpolm_w} B_x(m) A_x(n) B_x^*(m+n) \tag{2.6}$$

$$XPM_{4w} = \sum_{m,n} C_{m,n}^{xpolmw} B_x(m) A_y(n) B_y^*(m+n) \tag{2.7}$$

where B_x and B_y are the data symbols transmitted on X-pole and Y-pole of the channel w which is a neighbour to the base channel.

XPM noise samples have two main statistical characteristics:

- Space domain correlation
- Time domain correlation
- Correlation between X and Y poles



(a) Received data points

(b) After subtracting the long-term mean

Figure 2.3: XPM noise added to a 16QAM constellation points after passing through an ELEAF 25 span fiber

2.2.1 Space domain correlation

Received constellation points will be transferred and rotated as can be seen in Figure 2.3a . This plot shows that the shape of the clouds generated because of XPM noise around the constellation points are not circular. In other words, detection based on the minimum distance does not provide the best estimation of transmitted constellation points in this case. To find a better mode of detection, we need to fit a proper probability density function to each of the clouds.

For each of the constellation points, we find the principal components from eigenvalue decomposition. The angels of principal bases produce evidence that the joint probability density functions of real and imaginary parts of XPM samples have intrinsic information which can help us in detection.

2.2.2 Time Domain Correlation

Equations 2.4-2.7 Show a correlation in the time domain between the samples of XPM noise, coming from the intrinsic memory of nonlinear noise.

2.2.3 Correlation between X and Y poles

The physical properties of the fibers impel correlation between X-pole and Y-pole as it is evident in the XPM equation.

2.3 Detection using both space and time domain correlation

To exploit time and space domain correlation, instead of looking only at the current symbol, we will consider a block of three consecutive received data symbols for detection. In this case, we have to consider 4096 possible outcomes. If we consider these three complex numbers as 6 real numbers which are jointly Gaussian, we have to find joint probability density function for these six random variables conditional on all 4096 possible vectors. After calculating the covariance matrices and the average XPM noise vector for all of these possible vectors, we can maximize the equation 2.8 to detect the candidate with the highest conditional probability as the transmitted symbol.

$$P(\mathbf{Y}|\mathbf{S}) = \frac{1}{\sqrt{(2\pi)^6 det(\Sigma)}} exp\big[-\frac{1}{2}(\mathbf{Y}-\mu)^T\Sigma^{-1}(\mathbf{Y}-\mu)\big] \tag{2.8}$$

where $\mathbf{Y}$ is the received vector, $\mathbf{S}$ is one of the 4096 candidates, μ is the expected XPM noise vector for the candidate, and Σ is the covariance matrix of each of the candidates.

2.4 Results

To experiment detection using space and time domain features, we generate a stream of 2^{20} bits offline. The bits are modulated as 16 QAM. These symbols will pass a noisy channel which add just SPM noise to the symbols. Now, we can compute the covariance matrix for each candidate from 4096 possible combination of transmitted symbols. This process is done one more time with a difference that Gaussian noise is also added to the transmitted symbols. Finally, the received signals will be detected using equation 2.8.

As it can be seen in Fig , exploiting space and time domain correlation improves the bit error rate at high SNR values which Gaussian noise is not dominated. However, in low SNR values that SPM noise is masked by Gaussian noise, exploiting statistical characteristics of SPM noise fails to achieve a better result.

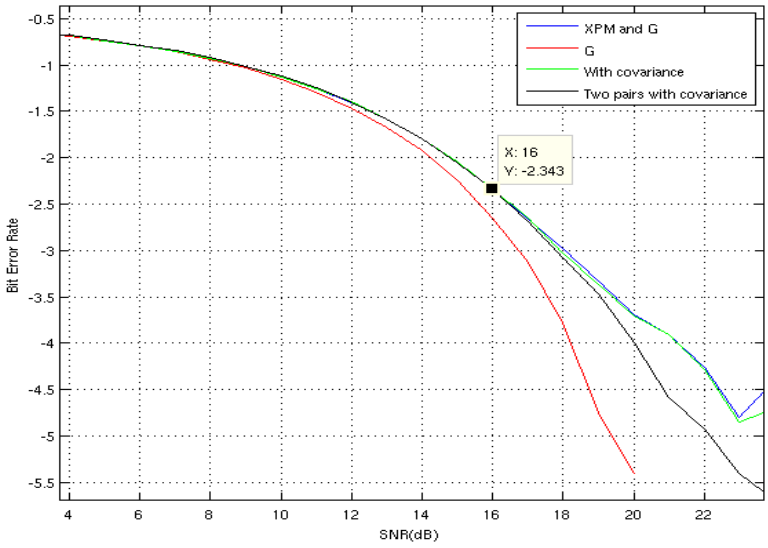


Figure 2.4: The effectiveness of exploiting time and space domain correlation

It should be noted that the variation of eigenvalues of the co-variance matrix if we consider channels x and y together, is almost 5 times bigger than the variation of eigenvalues of the co-variance matrix in time.

2.5 Summary

Chapter 2 focuses on extracting statistical characteristics of SPM and XPM noise. Based on the shape of clouds conditional on transmitted symbols, we can conclude that there is a significant correlation space domain. Also, our findings show that there is a correlation in time domain as well. So, SPM and XPM both have memory. Therefore, to exploit these charectristics we modify the naive minimum distance detection to consider not only the correlation between real and imaginary components of the noise, but also to consider the neighbour symbols.

2.5.1 Conclusions

The proposed method is useful only at high SNRs and does not improve the BER at the target SNR (around 13 dB) because the XPM noise is drown in Gaussian memory-less noise. It means the clouds of noise are almost spherical in space. However, using time domain correlation helped even in lower SNRs.

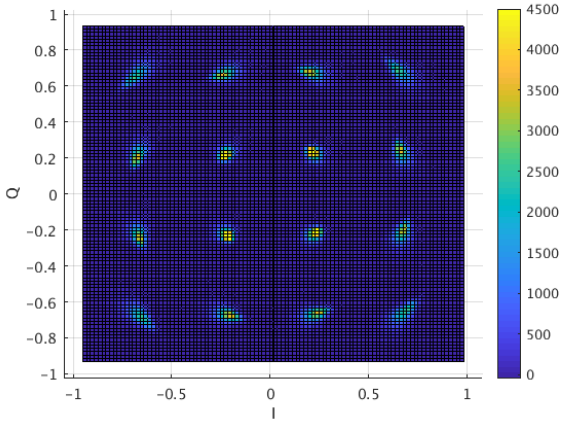
2.5.2 List of contributions

Our main contributions include

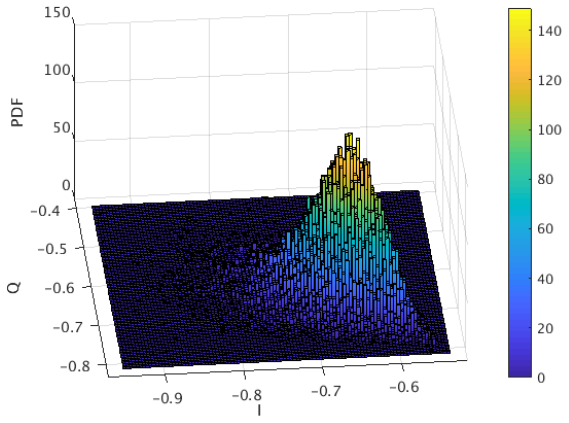
- Extracting some of statistical characteristics of SPM and XPM noise
- Proving the effectiveness of using these characteristics to improve detection

2.5.3 Future Research

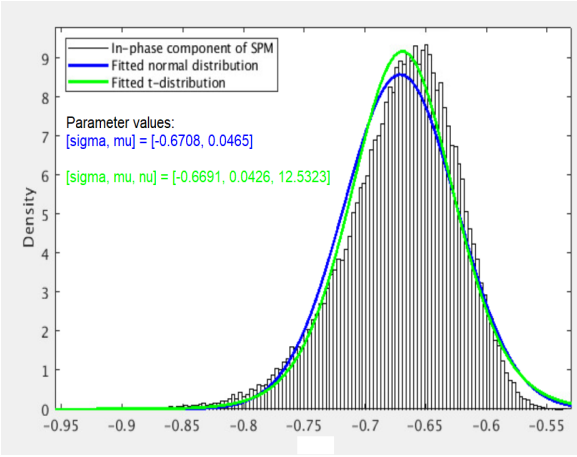
Let's take one step further to estimate the probability distribution of SPM more accurately. From the Fig 2.5 and 2.6 it can be seen that the SPM clouds around constellation points especially those which are farther from the origin do not look like a Gaussian distribution. Therefore, maybe fitting a better distribution to them would help to design a better detection method.



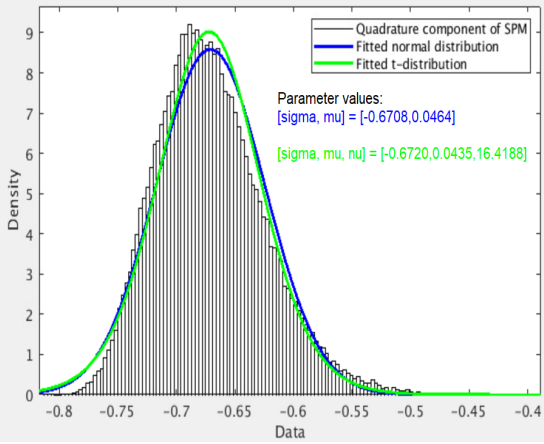
(a) PDF of the SPM noise



(b) PDF of the SPM noise around $A_x(t) = -0.67, -0.67$

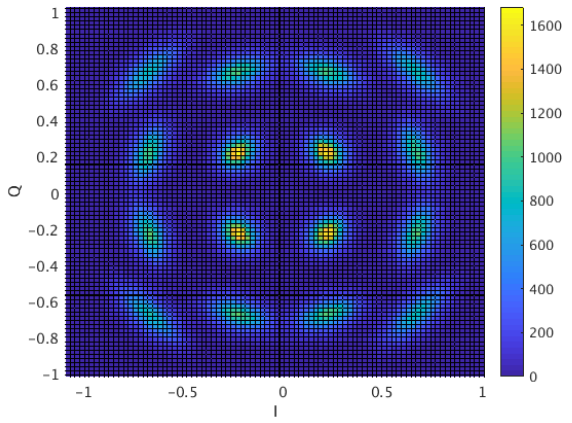


(c) In-phase component distribution around $A_x(t) = -0.67, -0.67$

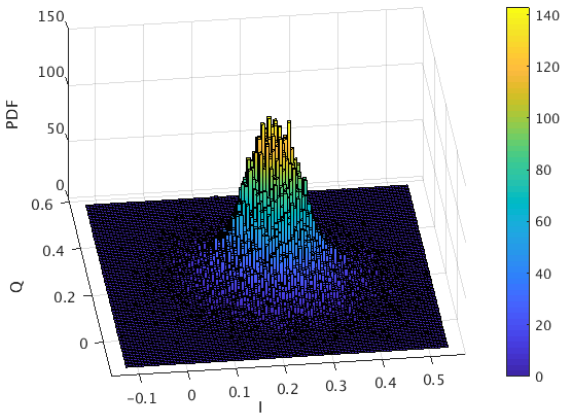


(d) Quadrature component distribution around $A_x(t) = -0.67, -0.67$ mean

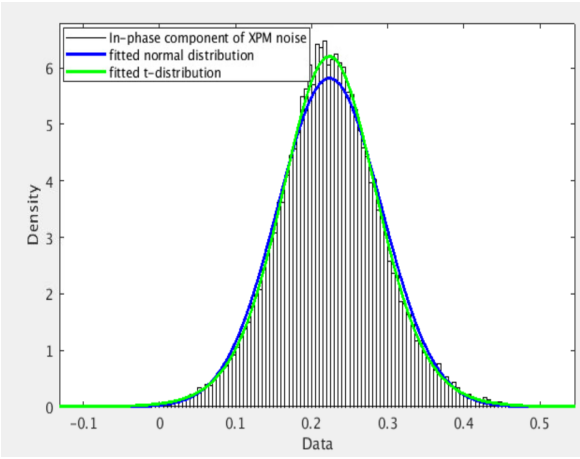
Figure 2.5: SPM noise distribution



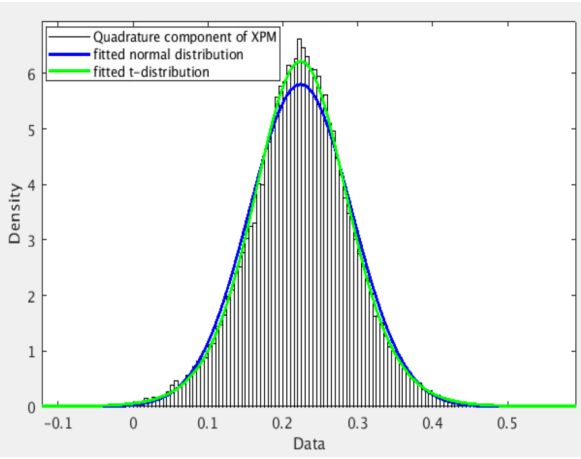
(a) PDF of the XPM noise



(b) PDF of the XPM noise around $A_x(t) = 0.23, 0.23$



(c) In-phase component distribution around $A_x(t) = 0.23, 0.23$



(d) Quadrature component distribution around $A_x(t) = 0.23, 0.23$ mean

Figure 2.6: XPM noise distribution

3

Iterative Decoding

I sent my Soul through the Invisible,
Some letter of that After-life to spell:
And by and by my Soul return'd to me,
And answer'd: 'I Myself am Heav'n and Hell

– Omar Khayyam, *Rubaiyat*

3.1 Introduction

In information theory, turbo codes, proposed in 1993 [16], are the first practical codes with near Shannon limit performance and low complexity in terms of decoding. While transmitting, the codes are constructed with different interleaved versions of two or more component codes. The component codes are decoded iteratively using two interconnected decoders at the receiver. In addition, a product code [23] is a data array of codewords. Each row and column represents a codeword from an (n1, k1) and (n2, k2) code, respectively. Product codes are also known as turbo product codes (TPCs) [24] because iterative (turbo) decoding is widely used to decode product codes. Simple parity check (SPC) based product codes with an iterative message-passing decoding algorithm show good performance in terms of bit-error rate (BER) [25]. This class of codes is called Product Accumulate (PA) codes.

We utilize this type of channel coding to compensate for nonlinear noise in the fiber-optic links for two main reasons. Firstly, iterative decoding at the receiver obviates the

need for feedback to the transmitter to pre-compensate for the SPM noise. Secondly, the decoding system can be modified to accept a priori information which helps us to remove nonlinear noise using demapping.

This chapter is organized as follows. In section 1, the coding structure is introduced. Section 2 shows how to decode the TPC codes iteratively. In section 3, the numerical results gained from 2 PAM and 16 QAM modulations, proving the effectiveness of demapping in nonlinear noise compensation, are presented.

3.2 System Model

The schematic structure of the system is shown in the Fig 3.1. First of all, raw data bits $\mathbf{u}$ are concatenated with P blocks of Simple Parity Check (SPC) code by a random inter-leaver π_2. Then the code passes an accumulator with rate 1 to generate the Product Accumulate (PA) code, $\mathbf{v}$. After encoding, modulated coded bits (s_k) pass through the channel which adds nonlinear noise (δA_x) and Amplified Spontaneous Emission (ASE) noise to the modulated coded bits. The received signal is decoded iteratively using two interconnected decoders at the receiver. The more detailed explanation of each component is as follows:

3.2.1 Encoding structure

The encoding structure is shown in Fig 1. Firstly, two branches of $(t+1, t)$ SPC codes generating P blocks of codewrods are concatenated with the raw input bits. After that, the TPC code is interleaved by a random interleaver π_2. The analysis shows that using a random interleaver provide us the opportunity to improve the performance of TPC/SPC codes [25]. Therefore, these codes have parameters of $(N, K, R) = (P(t+2), Pt, t/(t+2))$. Then the code passes an accumulator with rate-1 of the form 1/(1+D) to generate the Product Accumulate (PA) code. The idea of concatenating an outer code and an interleaver with a rate-1 recursive inner code, particularly of the form of 1/1+D, to achieve coding gains (interleaving gain) without reducing the overall code rate is widely recognized [25], [26].

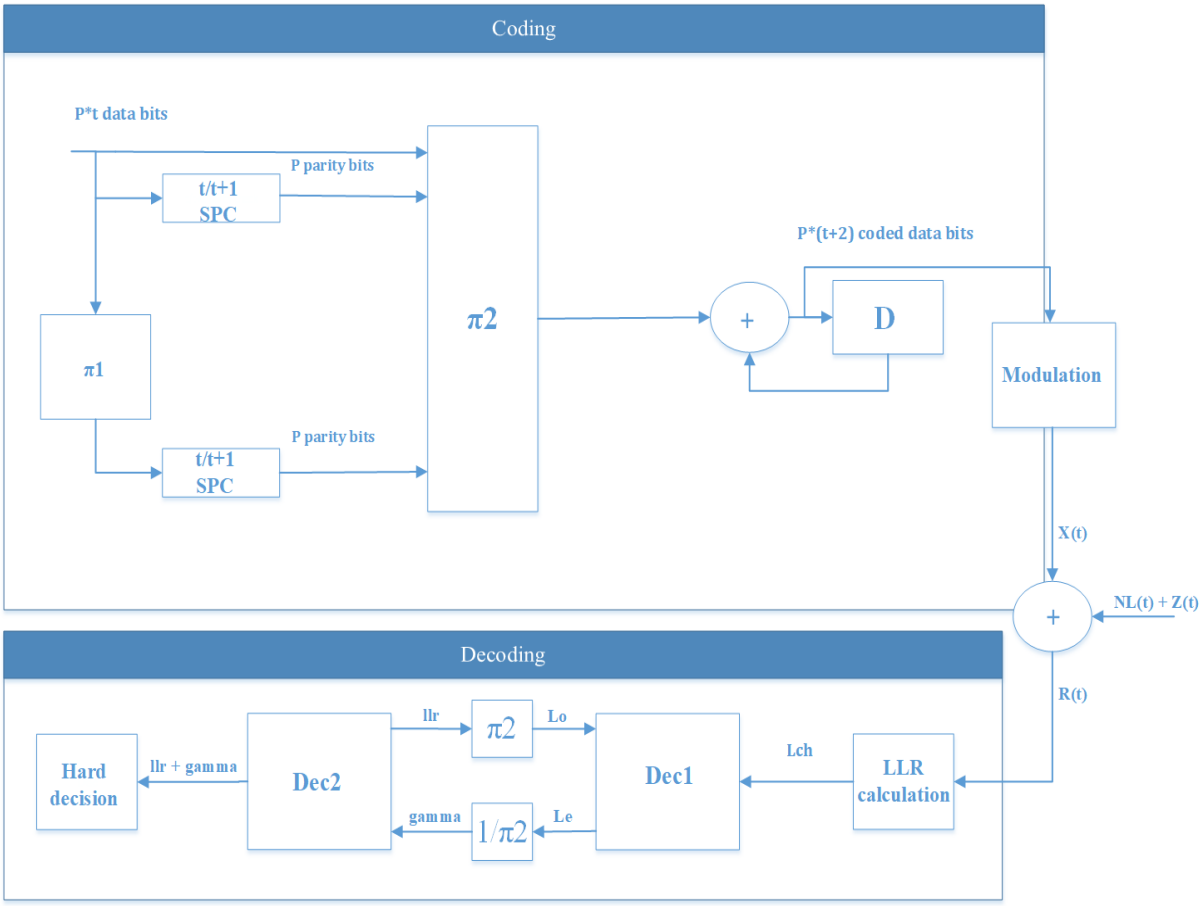


Figure 3.1: System model

3.2.2 Modulation

In this section, the 16QAM modulation of coded bits is explained which is easily extendable to 64QAM. We consider $s_k = s_{Ik} + js_{Qk}$ as a transmitted symbol with $s_{Ik}, s_{Qk} \in \{\pm 0.6708, \pm 0.23\}$. These values comes from the fact that the average power of the input signal to the fiber link has to be equal to 0.5 because the C matrix is designed in this way. According to the mapping rule $s_k = \mathcal{M}(b_{1k}, b_{2k}, b_{3k}, b_{4k})$, the information bits are mapped to transmitted symbols where the first two bits corresponds to the In-phase value $s_{Ik} = \mathcal{M}(b_{1k}, b_{2k})$ and the last two bits to the quadrature value $s_{Qk} = \mathcal{M}(b_{3k}, b_{4k})$.

After passing through the channel, the nonlinear noise plus an additive white Gaussian noise (AWGN) are added to the transmitted symbol which can be formulated as follows

$$r_k = r_{Ik} + jr_{Qk} = s_k + \delta A_x + n_k$$

where A_x is the nonlinear noise added to the X polarization of the channel according to the equation 2.1 and $n_k = n_{Ik} + jn_{Qk} = \mathcal{N}(0, N_0)$ is an AWGN where n_{Ik} and n_{Qk} have $\mathcal{N}(0, N_0/2)$ distribution.

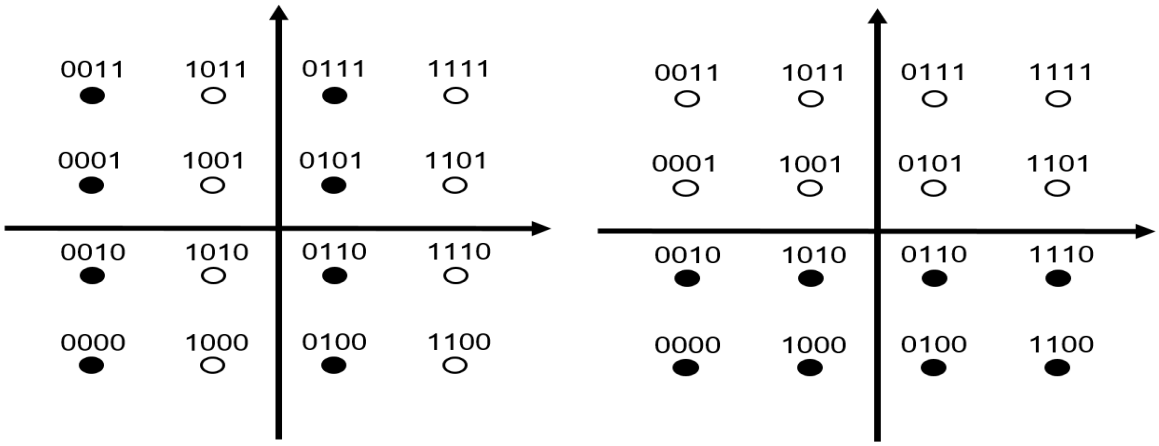
3.2.3 Decoding

A serially concatenated system is iteratively decodable using the turbo principle, in which the probability of bits is converted to log-likelihood ratio (LLR) values, which are called soft extrinsic information. As shown in Fig 2, these values are exchanged between the inner and outer decoder as a priori information iteratively. To decode the outer TPC/SPC code, a loopy belief propagation algorithm on graphical models is used. This algorithm calculates the marginal probability of each bit conditional on any other bits. The BCJR algorithm is typically used to decode the inner rate-1 convolutional code. For each bit x_i in the k_{th} turbo iteration, this algorithm produces extrinsic information, denoted L_e. L_e is used as a priori information by the outer decoder, and extrinsic information L_o is generated.

A. Log Likelihood Ratio Calculator

In this section the bit LLR value of the received signal L_{ch} is calculated for 16QAM which is easily extendable to other modulation types.

$$L_{ch}(b_{ik}|r_k) = log\frac{P(r_k|b_{ik} = 1)}{P(r_k|b_{ik} = 0)} \tag{3.1}$$



(a) Symbol set partitioning of b_{1k} (b) Symbol set partitioning of b_{4k}

Figure 3.2: Symbol set partitioning

Defining $S(b_{ik} = 1) = \{s : s = \mathcal{M}(..., b_{ik} = 1, ...)\}$ and $S(b_{ik} = 0) = \{s : s = \mathcal{M}(..., b_{ik} = 0, ...)\}$, the above equation can be calculated as

$$L_{ch}(b_{ik}|r_k) = log \frac{\sum\limits_{s_k \in S(b_{ik}=1)} P(r_k|s_k)}{\sum\limits_{s_k \in S(b_{ik}=0)} P(r_k|s_k)}$$

To illustrate the above equation, Fig 3.2 shows the symbol set partitioning for bits b_{1k} and b_{4k} [27]. We assume that the added noise can be model as an AWGN noise. Due to the Cartesian structure of the modulation, In-phase and quadrature components of the received signal can be demodulated independently which results in

$$L_{ch}(b_{1k}|r_k) = log\left\{ \frac{exp\big(-\frac{(r_{Ik}+a)^2}{N_0}\big) + exp\big(-\frac{(r_{Ik}-3a)^2}{N_0}\big)}{exp\big(-\frac{(r_{Ik}-a)^2}{N_0}\big) + exp\big(-\frac{(r_{Ik}+3a)^2}{N_0}\big)} \right\}$$

and

$$L_{ch}(b_{4k}|r_k) = log\left\{ \frac{exp\big(-\frac{(r_{Qk}-a)^2}{N_0}\big) + exp\big(-\frac{(r_{Qk}-3a)^2}{N_0}\big)}{exp\big(-\frac{(r_{Qk}+a)^2}{N_0}\big) + exp\big(-\frac{(r_{Qk}+3a)^2}{N_0}\big)} \right\}$$

B. Inner and Outer Decoders

According to [28], the inner convolutional code is decoded using a message passing algorithm is equivalent to BCJR algorithm for the 1/1+D code. This algorithm generates extrinsic information $L_e^{(k)}(x_i)$ for bit x_i in the k_{th} iteration accroding to equation 3.2 which is used as a priori information for the outer decoder to produce extrinsic information $L_o^{(k)}(x_i)$. In the next iteration, $L_o^{(k)}(x_i)$ and $L_{ch}^{(k)}(x_i)$ are used by the inner decoder to extract better estimation of the accumulator input bits.

$$L_e^{(k)}(x_i) = \textbf{check}\big(L_{ch}(y_i - 1) + L_{e_f}^{(k)}(y_i - 1), L_{ch}(y_i) + L_{e_b}^{(k)}(y_i)\big) \tag{3.2}$$

where

$$L_{ch}(y_i) = log\frac{P(r_i|y_i = 0)}{P(r_i|y_i = 1)}$$

is the LLR value of the received signal from the channel, $L_{e_f}^{(k)}(y_i)$ and $L_{e_b}^{(k)}(y_i)$ are the extrinsic information passed forward and backward to the bit y_i, respectively. The check operation is given as

$$\textbf{check}(\alpha, \beta) = 2tanh^{-1}\big(tanh\frac{\alpha}{2}tanh\frac{\beta}{2}\big)$$

In this work, we approximate it with

$$\textbf{check}(\alpha, \beta) = sign(\alpha).sign(\beta).min(|\alpha|, |\beta|)$$

which reduces required computations significantly [29]. During the last decoding iteration, the outer decoder generates LLRs for the input b_k, instead of LLRs for the coded symbols x_k. The LLRs for the input are then passed through a hard-decision device that decides on the decoded bits.

3.3 Short-Mean Updating

As we showed in 2.2, $A_x(t-1)$ and $A_x(t+1)$ influence the average of SPM added to the signal. Therefore, by estimating the neighbors of a received signal, we can compute the expected noise added by its neighbors and subtract it from the received signal. This the where we can take advantage of iterative decoding. After each iteration, we are improving

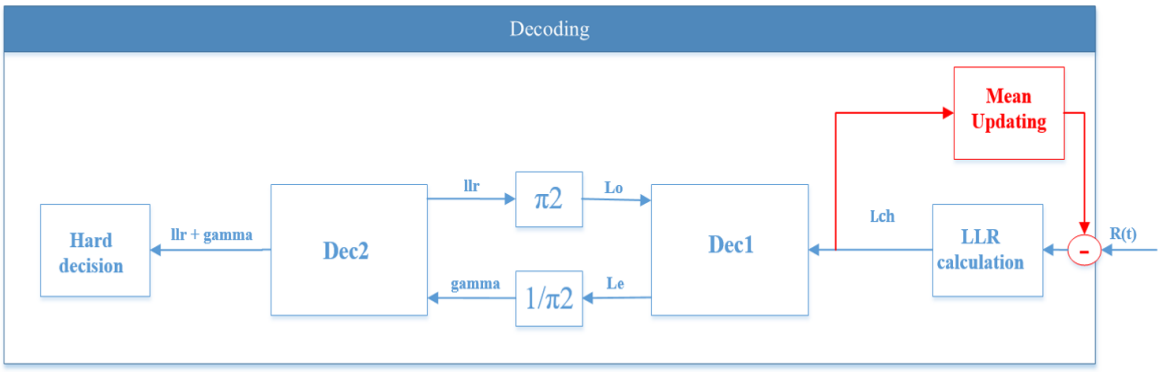


Figure 3.3: Updating short-term means

the estimation of transmitted symbols which can be used to compensate for SPM more efficiently. From the Eq. 3.1, it can be concluded that

$$L_{ch}(b_{ik}|r_k) = log\frac{P(b_{ik}=1|r_k)}{P(b_{ik}=0|r_k)} = log\frac{P(b_{ik}=1|r_k)}{1-P(b_{ik}=1|r_k)}$$

Therefore the APP values are computed as

$$P(b_{ik}=1|r_k) = \frac{e^{L_{ch}(b_{ik})}}{1+e^{L_{ch}(b_{ik})}}, P(b_{ik}=0|r_k) = \frac{1}{1+e^{L_{ch}(b_{ik})}}$$

Although the block random interleaver is used in single-parity-check (SPC) based product codes [30], it is shown in [28] that using random interleaver can improve the performance of these codes. Due to the random interleaving, we assume that coded bits are independent which results in

$$P(s_k) = P(b_{1k}, b_{2k}, b_{3k}, b_{4k}|r_k) = P(b_{1k}|r_k)P(b_{2k}|r_k)P(b_{3k}|r_k)P(b_{4k}|r_k)$$

Thus we calculate the probability of all possible combinations of three consecutive symbols as

$$P(s_{k-1}, s_k, s_{k+1}) = P(s_{k-1})P(s_k)P(s_{k+1})$$

Now using calculated short-term mean conditional on s_{k-1} and s_{k+1} in 2.2, we estimate the mean of SPM added to the s_k.

$$\mathbb{E}\left[SPM(s_k|s_{k-1}, s_{k+1})\right] = \sum_{i=1}^{16}\sum_{j=1}^{16}\sum_{k=1}^{16} P(s_{k-1}^i, s_k^j, s_{k+1}^k)\,\mathbb{E}\left[SPM(s_{k-1}^i, s_k^j, s_{k+1}^k)\right]$$

The modified decoding structure after adding short-term mean updating block is shown in Fig 3.3.

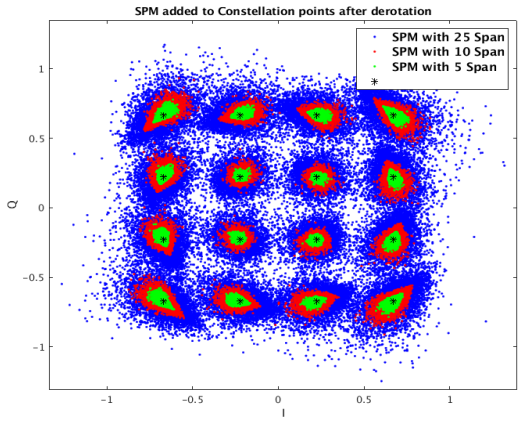
3.4 Results

We examined the proposed algorithm for both 16QAM and 64QAM. Fig 3.4 shows the immense improvement in 16 QAM modulation as a result of using iterative decoding instead of minimum distance detection. Moreover, it can be seen in Fig 3.5 that using iterative decoding with short-mean updating can improve the BER for both 5span and 10 span fiber. The results are reported for 6 and 10 numbers of iteration. Form the Fig 3.5b, it is evident that with the same number of iterations, using mean updating is even more effective for higher number of spans which is as expected (about 1dB improvement in BER for 10 spans compare to less than 0.1dB improvement for 5 spans).

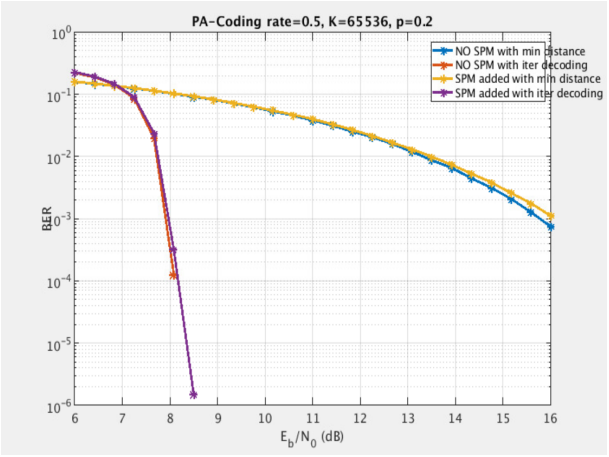
3.5 Summary and Conclusions

Chapter 3 focuses on using LDPC-Coded modulation with Iterative Damping and Decoding to improve bit probabilities. Firstly we described the system model including encoding structure, signal model and decoding structure. In section 3.3, we explained how to embed updating short-mean characteristic of SPM noise in iterative algorithm by adding a new block to the iterative decoding. Finally, we reported the experimental results in section 3.4 which prove the effectiveness of our proposed algorithm.

The iterative decoding system can be modified to accept a priori information (in our case short-term mean of nonlinear noise) which we proved can be used to remove nonlinear noise.

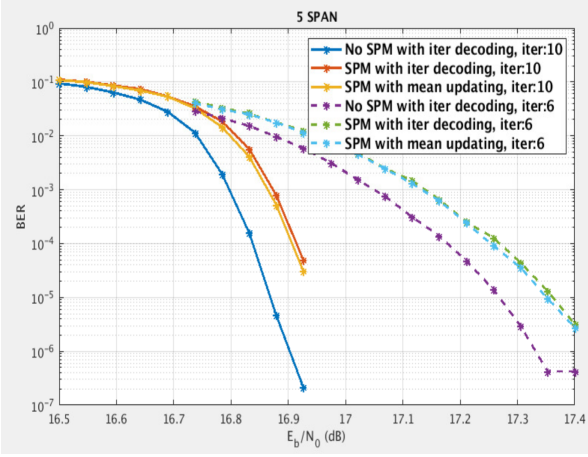


(a) After subtracting the long-term mean

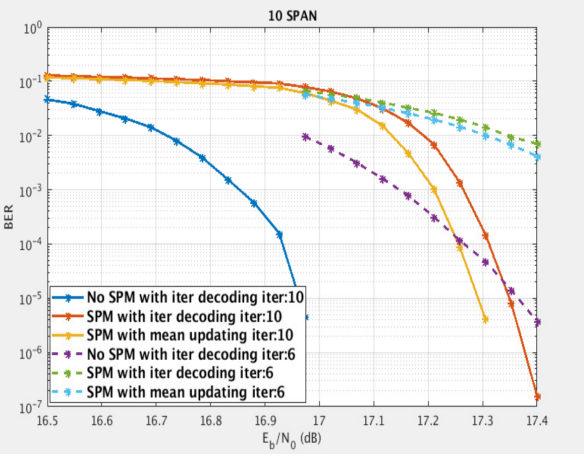


(b) The effectiveness of mean-updating in 5span, 10span and 25span fiber link

Figure 3.4: Experiments on 16QAM



(a) The effectiveness of mean-updating in 5spans fiber link



(b) The effectiveness of mean-updating in 10spans fiber link

Figure 3.5: Experiments on 64QAM

3.5.1 List of contributions

- Modifying the iterative decoding algorithm to accept short-term mean characteristic of nonlinear noise as a priori

4

Learning an Adaptive Model of Fiber

"Yesterday I was clever so I wanted to change the world. Today I am wise so I am changing myself."

– Rumi,

4.1 Introduction

Fiber non-linearity is one of the main bottlenecks toward increasing the capacity of the channel. However, some digital methods such as perturbation based pre-distortion (PPD) [31] [32] and digital back-propagation (DBP) [33] can be used to compensate for these impairments. PPD with 50 % chromatic dispersion pre-compensating not only can achieve a 2.4 dB in the relatice MSM on 800 kilometer of TWC fiber but also reduce the complexity significantly by quantizing the C-matrix coefficient [1]. However, fiber nonlinearty is dependant on several time-variant factors such as

- **Polarization mode dispersion (PMD):**

  Single mode fiber supports two degenerate polarization modes. Internal material stress, applied pressure, etc. break degeneracy by inducing different propagation constants along two principal polarization axes. This Phenomenon results in a differential group delay (DGD) between polarization modes. Commonly referred to as first order PMD.

- **Polarization dependent loss (PDL):** Optical elements (couplers, WSS, EDFAs) often exhibit some degree of loss that depends on the state of input polarization.
- **Non-uniform power of each span:** In order to calculate total C-matrix for more than one span, one needs to compute weighted sum of each span's C-matrix with the output power of each span as its weight. This is not possible practically because the output power of each span is not deterministic. Learning algorithm can handle this issue by updating the fiber's model.

In this chapter, we propose a pre-compensation algorithm shown in Fig 4.1. However the main advantage of our algorithm is that instead of calculating C-matrix analytically, we learn C-matrix based on the inputs and outputs of the channel which makes our algorithm adaptive to different situations of the fiber. Then, we apply the same complexity reduction methods as [1] to have a fair comparison in terms of performance and complexity. Furthermore, this method is easily extendable to learn higher order terms of the fiber's model which analytic C-matrix is not capable of. This chapter is organized as follows. In section 4.2, we introduce the learning procedure and extending it to higher order parameters in section 4.2.1. In section 4.2.2, we discuss some techniques to reduce the complexity of the learning algorithm. finally, we compare the performance and complexity of the learning algorithm with the reference pre-compensation method [1] in section 4.3

4.2 Model

A single channel fiber can be modeled as

$$r_x = A_x + \delta A_x + n_x$$

where r_x and A_x are received and transmitted symbols on polarization X. δA_x is the nonlinear noise added according to the equation 2.1 and n_x is the Gaussian noise. Encapsulating triplets of symbols as a term, equation 2.3 can be written as

$$SPM = \sum_{m,n} C^{spm}_{m,n} A_x(m)\big[A_x(n)A^*_x(m+n) + A_y(n)A^*_y(m+n)\big] = \sum_{m,n} C^{spm}_{m,n}\mathbf{A}_{m,n}$$

If we flatten a matrix of triplets as a row

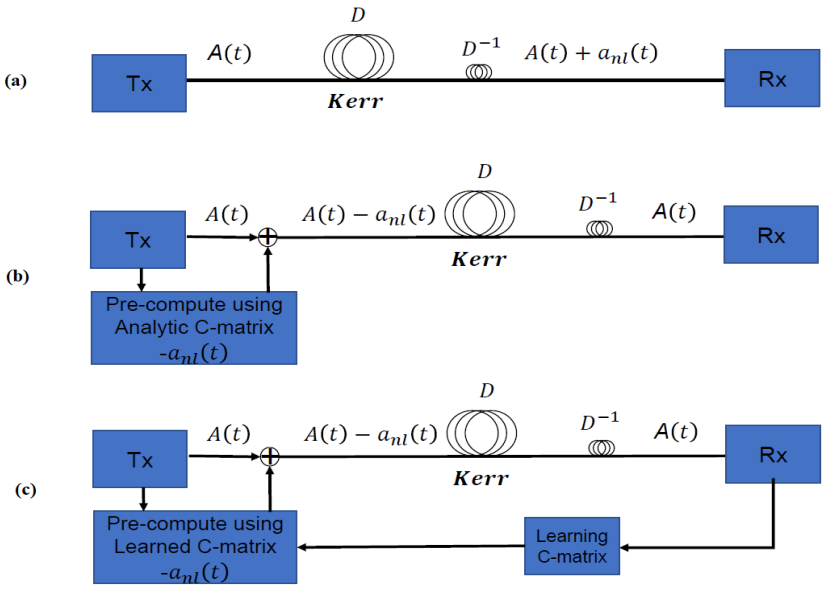


Figure 4.1: (a) Standard (b) Pre-compensation using Analytic C-matrix [1] (c) Our Algorithm

$$\begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \dots & \mathbf{A}_{1d} \\ \mathbf{A}_{1(d+1)} & \mathbf{A}_{2(d+2)} & \dots & \mathbf{A}_{1(2d)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{1,(n-1)d} & \mathbf{A}_{1,(n-1)d+1} & \dots & \mathbf{A}_{1d^2} \end{pmatrix} \Longrightarrow \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \dots & \mathbf{A}_{1d^2} \end{pmatrix}$$

Rewriting the equation 4.1 for n number of samples, we can derive a linear system of equations between triplets of transmitted symbols and received SPM noise.

$$\begin{pmatrix} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \dots & \mathbf{A}_{1,d^2} \\ \mathbf{A}_{2,1} & \mathbf{A}_{2,2} & \dots & \mathbf{A}_{2,d^2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{t,1} & \mathbf{A}_{t,2} & \dots & \mathbf{A}_{t,d^2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{n,1} & \mathbf{A}_{n,2} & \dots & \mathbf{A}_{n,d^2} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{d^2} \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_n \end{pmatrix}$$

or equivalently

$$\mathbf{A} \times C_L = \mathbf{S} \tag{4.1}$$

where

$$A_{tj} = A_x[t+m]A_x[t+n]A_x^*[t+m+n] + A_x[t+m]A_y[t+n]A_y^*[t+m+n] \tag{4.2}$$

Equation 4.1 is a system of linear equations which is overdetermined. This can be translated to minimizing square-norm of distance between the SPM values and the output of our model or

$$\min_{c_1,\ldots,c_{d^2}} \quad ||\mathbf{S} - \mathbf{A}C_L||^2 \tag{4.3}$$

where $\mathbf{S}$ is the observed output (SPM noise + Gaussian noise), $\mathbf{A}$ is the matrix of triplets, and C_L is the vector of learned coefficients. We can find parameters of C-matrix by multiplying pseudo-inverse of matrix $\mathbf{A}$ with observed noise vector.

$$C_L = \mathbf{A}^+\mathbf{S} \tag{4.4}$$

where $\mathbf{A}^+ = (\mathbf{A}^*\mathbf{A})^{-1}\mathbf{A}$ and $\mathbf{A}^*$ is hamiltonian transpose of matrix $\mathbf{A}$. Now we have an estimate of C-matrix.

4.2.1 2D C-matrix to 3D C-matrix

In reference [34] and [35], a general expression of the nonlinear impairments in an optical fiber is developed which provides a more comprehensive model compared to the equation 2.3. Therefore, we can update the equation of the SPM noise as

$$SPM = \sum_{l,m,n} C_{l,m,n}^{spm} A_x(m)\left[A_x(n)A_x^*(m+n+l) + A_y(n)A_y^*(m+n+l)\right] \tag{4.5}$$

One of the advantages of the learning algorithm proposed in this work is that it can be easily extended to include higher order terms of the equation 4.5 for $l = \pm 1$. Thus, the equation 4.1 can be modified as

$$\begin{pmatrix} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \ldots & \mathbf{A}_{1,d^2} \ldots\ldots \mathbf{A}_{1,3d^2} \\ \mathbf{A}_{2,1} & \mathbf{A}_{2,2} & \ldots & \mathbf{A}_{2,d^2} \ldots\ldots \mathbf{A}_{2,3d^2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{t,1} & \mathbf{A}_{t,2} & \ldots & \mathbf{A}_{t,d^2} \ldots\ldots \mathbf{A}_{t,3d^2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{n,1} & \mathbf{A}_{n,2} & \ldots & \mathbf{A}_{n,d^2} \ldots\ldots \mathbf{A}_{n,3d^2} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{d^2} \\ \vdots \\ \vdots \\ c_{3d^2} \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_n \end{pmatrix}$$

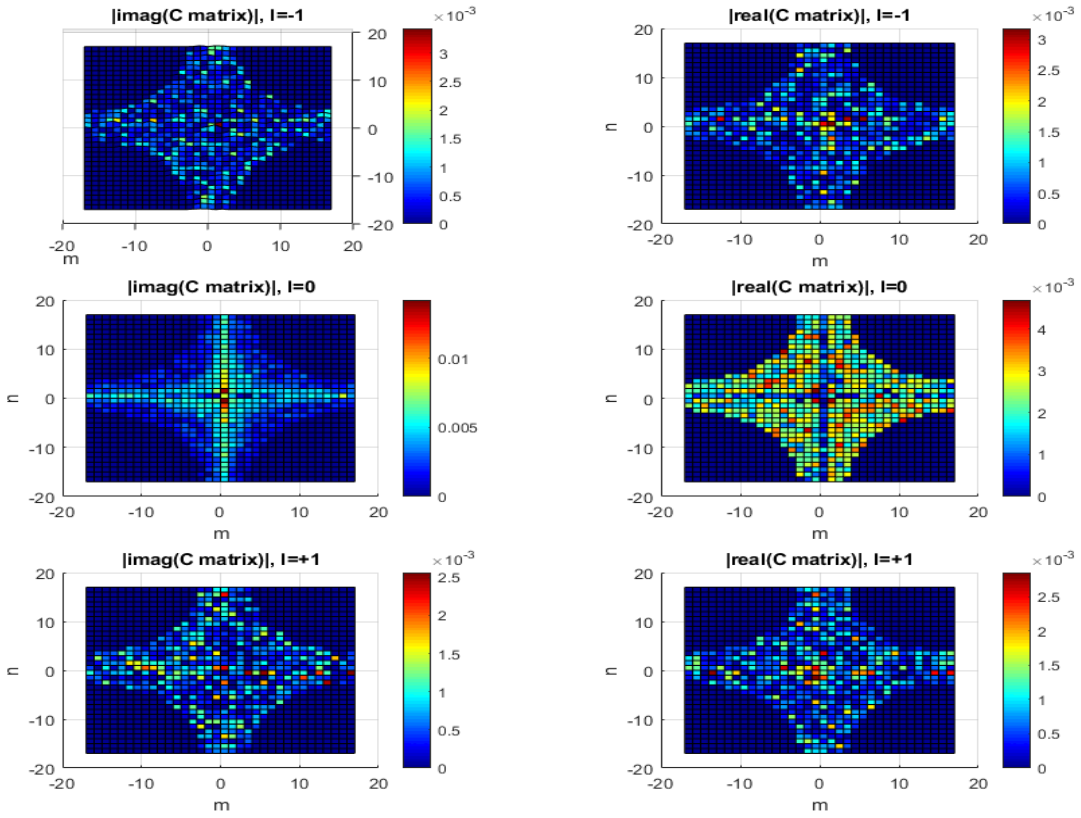


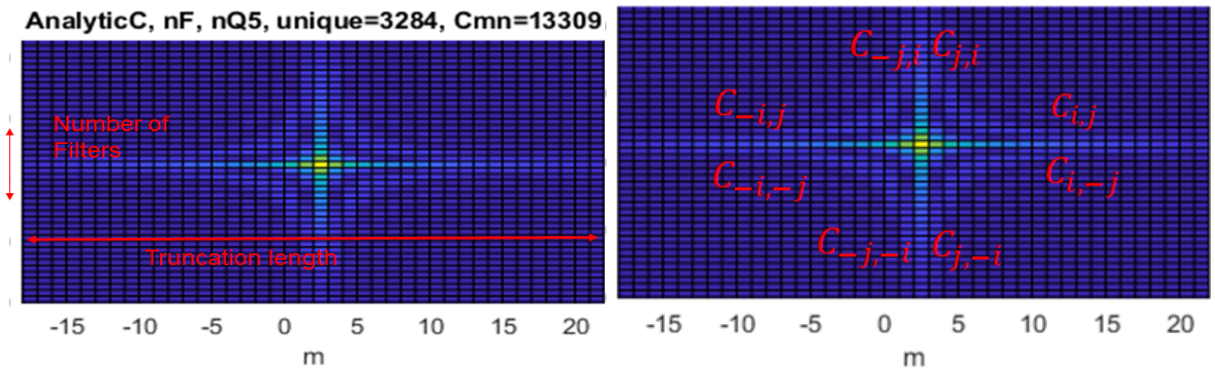
Figure 4.2: Learned C-matrix for $l = 0, \pm 1$

Where $\{c_i\}_{i=1}^{d^2}$ are corresponding to the zero order C-matrix and $\{c_i\}_{i=d^2+1}^{3d^2}$ are composing higher order C-matrices.

Fig 4.2 shows the absolute values of the real and imaginary components of the learned C matrices.

4.2.2 Complexity Reduction

The complexity of pre-compensation method is roughly proportionate to the total number of C-matrix coefficients and unique coefficients. To reduce these factors some methods are used in [1] such as quantization and filtering on only some rows and columns of C-matrix.



(a) Parameteres determining the model complexity (b) Forcing symmertries

Figure 4.3: Efficient learning of C-matrix

Therefore, as shown in Fig 4.5a, the number unique of coefficients is dependent on three parameteres: 1- Truncation length, 2- Number of horizontal and vertical filters, 3- Number of quantization levels.

Since we know that nonlinear noise forces the C-matrix to have symmetries, we can leverage this fact to reduce the complexity of the learning algorithm. For this purpose, several columns of matrix **A** which their corresponding C values have symmetry (Fig 4.3b), can be summed up. This technique can reduce the running time of the learning algorithm up to the factor of 8.

Moreover, it can be seen in the analytical C-matrix that its coefficients diminishes as they get farther from the center. Therefore, limiting the learning coefficients to those around the center and middle axes do not deteriorate the learning results significantly. To this end, we defined two hyperbolic functions. Our algorithm Only learns those coefficients inside these two functions and ignores others. This will add a hyper-parameter to our learning algorithm which will be optimized later.

After fitting the model, we quantize learned coefficients and remove those ones that are very small relative to highest coefficient to reduce the number of unique coefficients which is one of the main factors determining the complexity of our algorithm. Fig 4.4 shows both analytic and learned C-matrix for number of filters (nF) equal to 3, 19 and infinity(Which means full complexity mode). Evidently, the number of coefficients needed for learning the C-matrix is far less than the ones calculated with analytical approach.

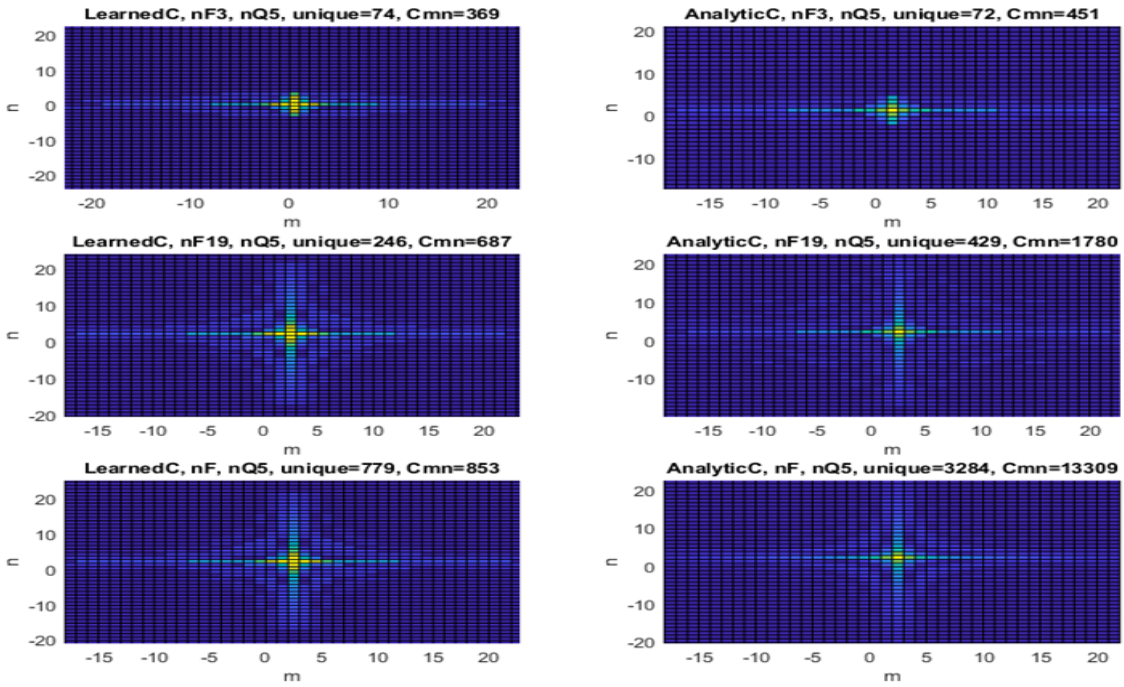
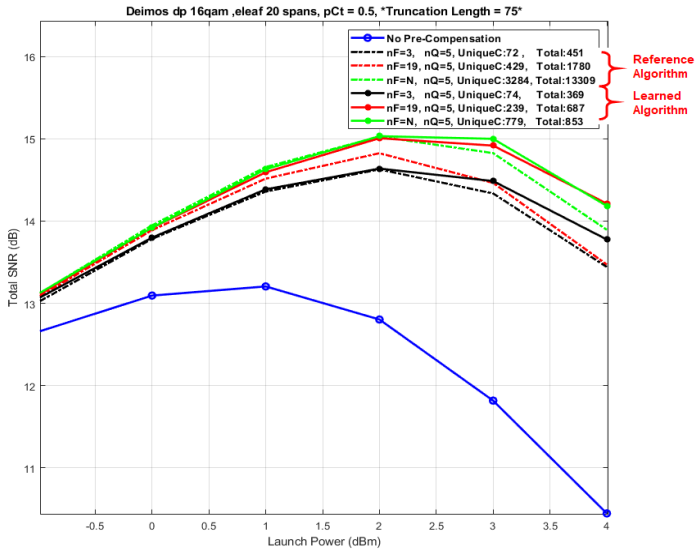


Figure 4.4: Learned vs Analytic C-matrix



(a) Performance of the proposed algorithm compare to the reference algorithm

Figure 4.5: Efficient learning of C-matrix

4.3 Results

We did experiments for different types of fibers like NDSF, ELEAF and TWC. In lower launch power, nonlinear noise is not significant compare to ASE noise and increasing the launch power helps to have a higher total SNR. However, nonlinear noise becomes dominant in higher launch power and as a result total SNR fall. Therefore, there is a optimum launch power which lead to the highest SNR. As can be seen in the the Fig 4.5, learning algirthm results in higher total SNR for higher launch power while keeping the number of coefficients smaller. Fig 4.5 shows the effectiveness of the learning C-matrix for different level of complexity reduction. These results opens up a huge opportunity for further research on designing more accurate and efficient learning algorithms which would be capable of higher gain in performance and complexity for pre-compensation methods.

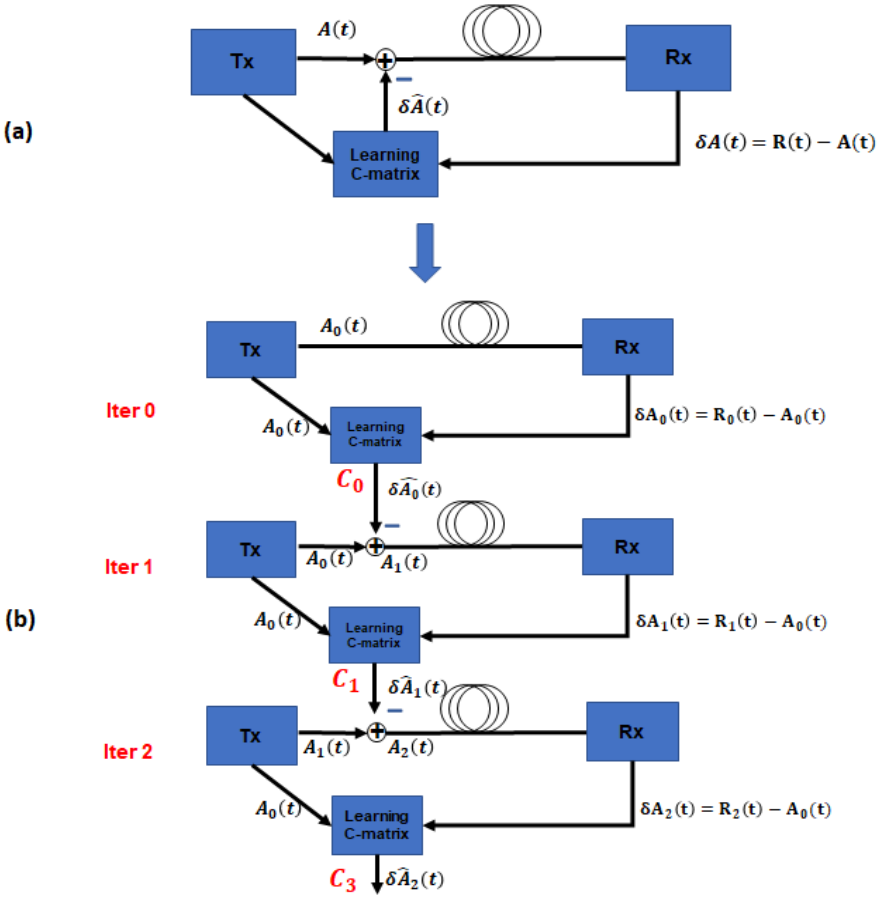


Figure 4.6: (a) Standard (b) unrolled structure

4.3.1 Iterative Pre-Compensation

We proved that one step pre-compensation improve the total SNR significantly. Now, we propose to repeat this algorithm more than one iteration to pre-compensate the residuals of SPM noise resulting from the first iteration. In other words, by learning a new C-matrix from the pre-compensated input (A_x) and the residual of the SPM noise after iteration one (δA_2) , we expect to be able to model higher order terms in nonlinear noise. The structure of the iterative pre-cmpensation is shown in the Fig 4.6 which can be formulated as follows for N number of iteration

$$A_{i+1} = A_i - \mathbf{A}C_i, i = 0, ..., N$$

where $\mathbf{A}$ is the matrix of triplets (4.1) and C_i is the learned C-matrix at iteration i which is a function of the original transmitted symbols A_0 and the SPM noise at iteration i:

$$C_i = \text{pinv}(\mathbf{A}) \times \delta A_i$$

where

$$\delta A_i = R_i - A_0, i = 0, ..., N$$

where A_i is the stream of data transmitted at iteration i and δA_i is the stream of noise at iteration i. We expect δA_i to approach zero as we repeat this algorithm.

The results of the iterative pre-compensation for both ELEAF and NDSF applications with 16 QAM modulation and 50% inline pre-compensation are reported in Fig 4.7 and 4.8, respectively. Truncation length for all C-matrices used for both Analytical Learning pre-compensation is fixed to 55 and hyper Constant for learning Algorithm is fixed to 80. As it can be seen in both reduced and Full complexity mode, adding higher order C-matrices ($L = \pm 1$) has improved the total SNR but its improvement is not significant comparing to the number of coefficients which it adds to the algorithm. On the other hand, repeating pre-compensation for the $2nd$ iteration helps significantly especially in full complexity mode. However, the reduced complexity version of the algorithm does not perform better at the $2nd$ iteration for NDSF fiber. This might be caused by the fact that we use the same complexity reduction techniques in both iterations. The structure of the C-matrix at higher iterations is not necessarily same as iteration one. So, it might need a different tricks to reduce its complexity which will be pursued in the future research.

complexity mode, adding higher order C-matrices ($L = \pm 1$) has improved the total SNR but its improvement is not significant comparing to the number of coefficients which it adds to the algorithm. On the other hand, repeating pre-compensation for the $2nd$ iteration

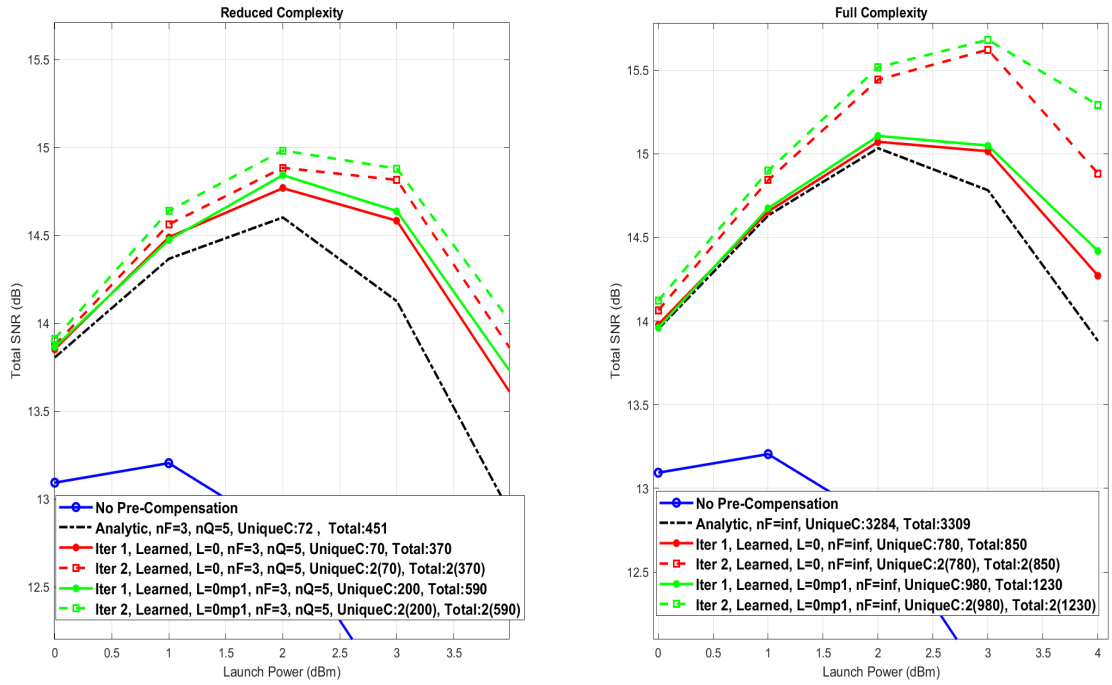


Figure 4.7: Deimos dual polarization 16QAM, eleaf 20 spans, pCt = 0.5, Truncation Length = 55

helps significantly especially in full complexity mode. However, the reduced complexity version of the algorithm does not perform better at the $2nd$ iteration for NDSF fiber. This might be caused by the fact that we use the same complexity reduction techniques in both iterations. The structure of the C-matrix at higher iterations is not necessarily same as iteration one. So, it might need a different tricks to reduce its complexity which will be pursued in the future research.

4.3.2 Hyper-parameter optimization

The proposed algorithm has two sets of hyper-parameters C-matrices dimensions $\{dimC_i\}_{i=-1}^{1}$ and hyperbolic constant $\{H_i\}_{i=-1}^{1}$ which can be optimized. Fig 4.9 shows the success rate of the learning algorithm to estimate the SPM noise as a function of $dimC$ and H. The success rate is increasing by the $dimC$ and H. However, bigger C-matrix means higher

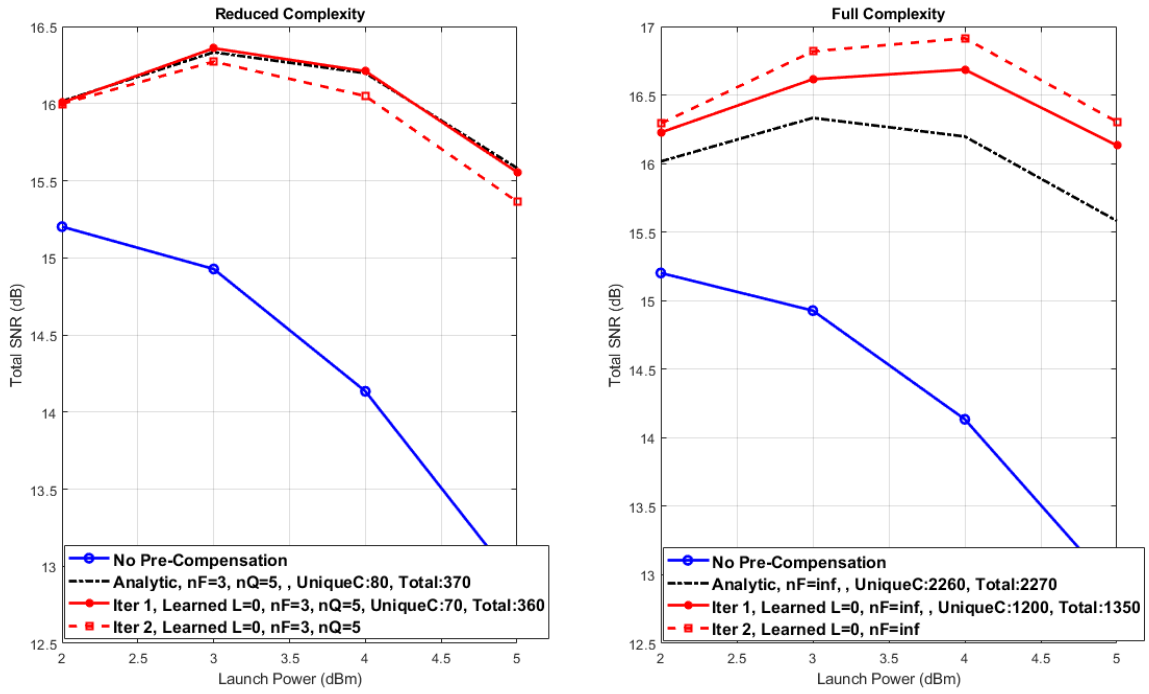


Figure 4.8: Deimos dual polarization 16QAM, ndsf 20 spans, pCt = 0.5, Truncation Length = 55

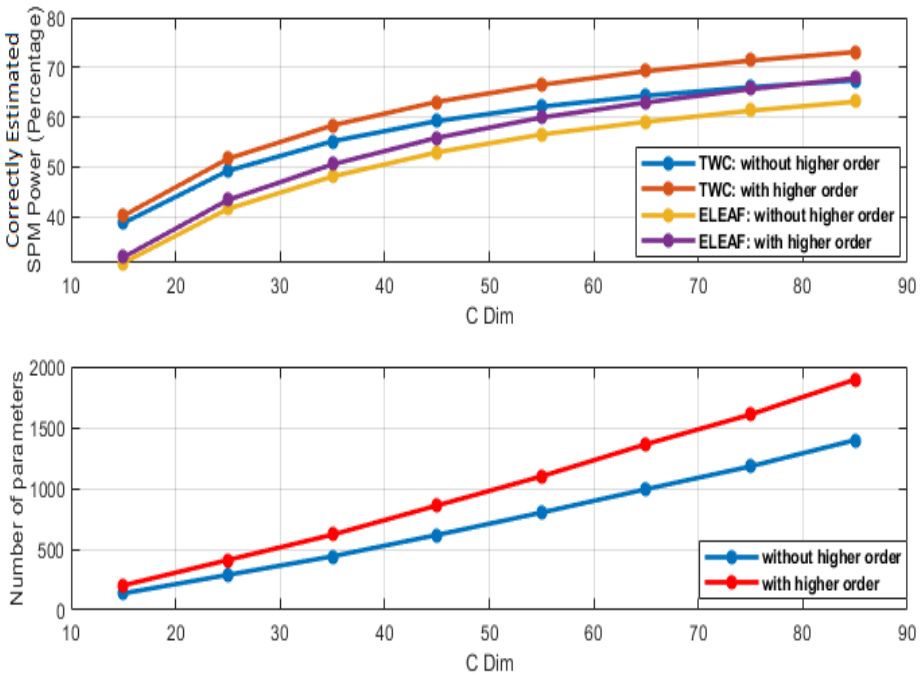


Figure 4.9: SPM estimation

number of coefficients which is not desirable as determining factor of algorithm complexity. Therefore, we need to fine tune these parameters to maximize the algorithm performance while keeping the complexity low. Fig 4.10 reports total SNR after first and second iteration of pre-compensation for truncation length of 15, 25, 35, 55, 75 with Hyper constants of 20, 50, 80. The number of unique coefficients which is a key factor in determining the complexity of the algorithm is reported in red color for some of these combinations.

4.4 Summary

In this chapter, we introduced a pre-compensation algorithm which instead of analytic calculation, learns C-matrix based on the inputs and outputs of the channel. This makes the algorithm robust to different situations of the fiber. Then, we applied the same complexity reduction methods as [1] to have a fair comparison in terms of performance and complexity. The other advantage of this method, which is explained in section 4.2.1, is the flexibility

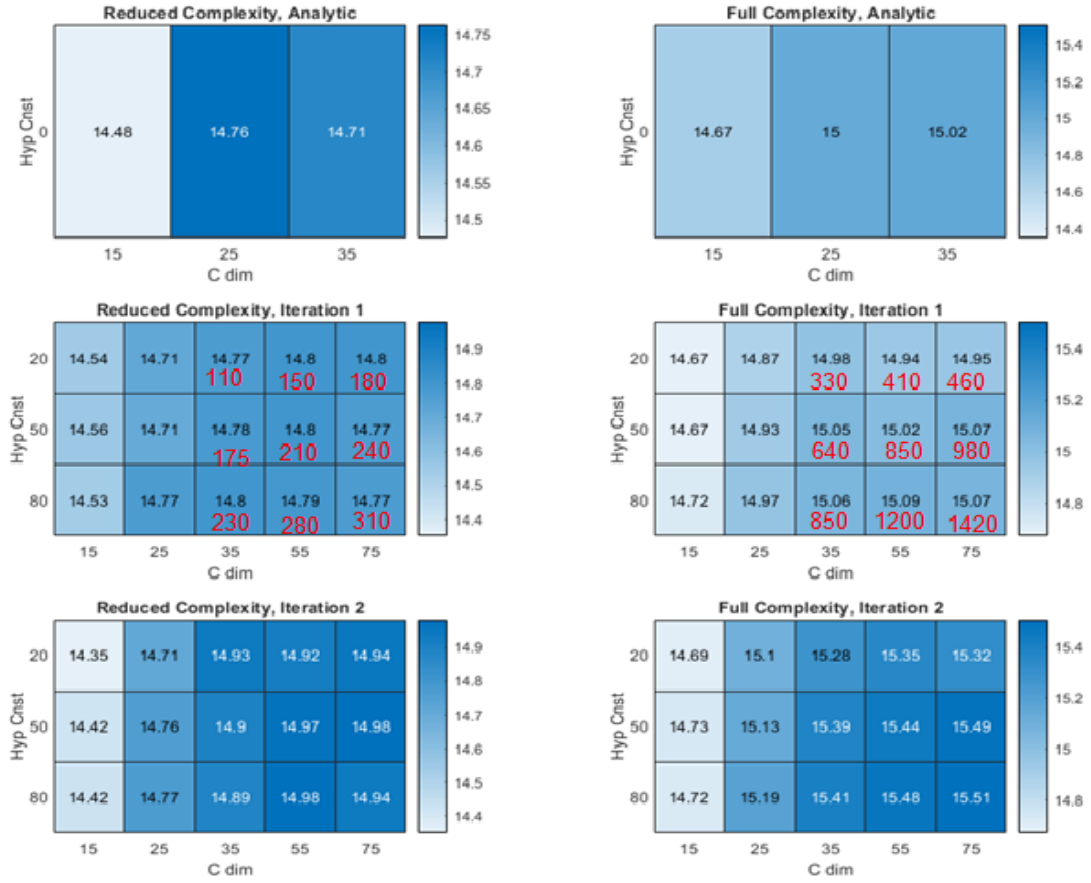


Figure 4.10: Deimos dual polarization 16QAM, eleaf 20 spans, pCt = 0.5, Truncation Length = 55

of learning higher order terms which analytic approach is not capable of.

4.4.1 Conclusions

Learning C-matrix not only make pre-compensation robust to many variations in factors affecting the nonlinear noise, but also improves the performance of while reducing the complexity of the pre-compensation method. Moreover, this approach gives us the felexibilty of adding higher order terms to estimate the nonlinear noise. Finally, to find the best C-matrix underlying the fiber-optic systems, we need to rely on iterative pre-compensation

to mitigate the non-linear noise using a linear approach.

4.4.2 List of contributions

- Proposing an adaptive model of fiber for pre-compensation of nonlinear noise
- Adding high-order terms to the noise calculation
- Proving the effectiveness of iterative pre-compensation empirically to model higher order nonlinearities
- Proposing several complexity reduction techniques

4.4.3 Future research

The proposed learning algorithm can be even further optimized to achive the best performance while reducing the complexity by finding a lower space representation of C-matrices filters. More importantly, we restricted ourselves to triplets of symbols to calcuate SPM noise. We can relax this assumption or use a deep neural network (DNN) or recurrent neural network (RNN) to find the optimum model of the fiber. We decided to implement the iterative pre-compensation by subtracting the estimated values of the nonlinear noise directly from the input because of its simplicity and low complexity. In future research, a more correct model would be calculating the Jacobin matrix of the output of the fiber with respect to the each one of the coefficients of the C-matrix and apply gradient descent.

References

[1] Qunbi Zhuge, Michael Reimer, Andrzej Borowiec, Maurice O'Sullivan, and David V. Plant. Aggressive quantization on perturbation coefficients for nonlinear predistortion. In *Optical Fiber Communication Conference*, page Th4D.7. Optical Society of America, 2014.

[2] Linn F Mollenauer, Stephen G Evangelides, and James P Gordon. Wavelength division multiplexing with solitons in ultra-long distance transmission using lumped amplifiers. *Journal of lightwave technology*, 9(3):362–367, 1991.

[3] Akira Hasegawa, Shiva Kumar, and Yuji Kodama. Reduction of collision-induced time jitters in dispersion-managed soliton transmission systems. *Opt. Lett.*, 21(1):39–41, Jan 1996.

[4] Keang-Po Ho and Joseph M Kahn. Electronic compensation technique to mitigate nonlinear phase noise. *Journal of Lightwave Technology*, 22(3):779, 2004.

[5] Hoon Kim and Alan H Gnauck. Experimental investigation of the performance limitation of dpsk systems due to nonlinear phase noise. *IEEE Photonics Technology Letters*, 15(2):320–322, 2003.

[6] Antonio Mecozzi. Limits to long-haul coherent transmission set by the kerr nonlinearity and noise of the in-line amplifiers. *Journal of lightwave technology*, 12(11):1993–2000, 1994.

[7] Keang-Po Ho. *Phase-modulated optical communication systems*. Springer Science & Business Media, 2005.

[8] Govind P Agrawal. Nonlinear fiber optics. In *Nonlinear Science at the Dawn of the 21st Century*, pages 195–211. Springer, 2000.

[9] Saheb Pasand, Ali. Methods for nonlinear impairments compensation in fiber-optic communication systems, 2018.

[10] Liang Bangyuan Du and Arthur J Lowery. Practical xpm compensation method for coherent optical ofdm systems. *IEEE Photonics Technology Letters*, 22(5):320–322, 2010.

[11] William Shieh, Hongchun Bao, and Yan Tang. Coherent optical ofdm: theory and design. *Optics express*, 16(2):841–859, 2008.

[12] Xiaoxu Li, Xin Chen, Gilad Goldfarb, Eduardo Mateo, Inwoong Kim, Fatih Yaman, and Guifang Li. Electronic post-compensation of wdm transmission impairments using coherent detection and digital signal processing. *Optics Express*, 16(2):880–888, 2008.

[13] Ezra Ip and Joseph M Kahn. Compensation of dispersion and nonlinear impairments using digital backpropagation. *Journal of Lightwave Technology*, 26(20):3416–3425, 2008.

[14] Giulio Colavolpe, Gianluigi Ferrari, and Riccardo Raheli. Extrinsic information in iterative decoding: A unified view. *IEEE Transactions on Communications*, 49(12):2088–2094, 2001.

[15] J Lodge, R Young, P Hoeher, and J Hagenauer. Separable map" filters" for the decoding of product and concatenated codes. In *Proceedings of ICC'93-IEEE International Conference on Communications*, volume 3, pages 1740–1745. IEEE, 1993.

[16] Claude Berrou, Alain Glavieux, and Punya Thitimajshima. Near shannon limit error-correcting coding and decoding: Turbo-codes. 1. In *Proceedings of ICC'93-IEEE International Conference on Communications*, volume 2, pages 1064–1070. IEEE, 1993.

[17] Lalit Bahl, John Cocke, Frederick Jelinek, and Josef Raviv. Optimal decoding of linear codes for minimizing symbol error rate (corresp.). *IEEE Transactions on information theory*, 20(2):284–287, 1974.

[18] Stephan Ten Brink, Joachim Speidel, and Ran-Hong Yan. Iterative demapping and decoding for multilevel modulation. In *IEEE GLOBECOM 1998 (Cat. NO. 98CH36250)*, volume 1, pages 579–584. IEEE, 1998.

[19] Patrick Robertson. Illuminating the structure of code and decoder of parallel concatenated recursive systematic (turbo) codes. In *1994 IEEE GLOBECOM. Communications: The Global Bridge*, volume 3, pages 1298–1303. IEEE, 1994.

[20] Catherine Douillard, Michel Jézéquel, Claude Berrou, Département Electronique, Annie Picart, Pierre Didier, and Alain Glavieux. Iterative correction of intersymbol interference: Turbo-equalization. *European transactions on telecommunications*, 6(5):507–511, 1995.

[21] Hosein Nikopour, Amir K Khandani, and Seyed Hamidreza Jamali. Turbo-coded ofdm transmission over a nonlinear channel. *IEEE transactions on vehicular technology*, 54(4):1361–1371, 2005.

[22] Irshaad Fatadin, David Ives, and Seb J. Savory. Blind equalization and carrier phase recovery in a 16-qam optical coherent system. *J. Lightwave Technol.*, 27(15):3042–3049, Aug 2009.

[23] Ramesh Pyndiah, Alain Glavieux, Annie Picart, and Sylvie Jacq. Near optimum decoding of product codes. In *1994 IEEE GLOBECOM. Communications: The Global Bridge*, pages 339–343. IEEE, 1994.

[24] Ramesh Mahendra Pyndiah. Near-optimum decoding of product codes: Block turbo codes. *IEEE Transactions on communications*, 46(8):1003–1010, 1998.

[25] Jing Li, Krishna R Narayanan, and Costas N Georghiades. Product accumulate codes: a class of codes with near-capacity performance and low decoding complexity. *IEEE Transactions on Information Theory*, 50(1):31–46, 2004.

[26] D Divsalar and F Pollara. Serial and hybrid concatenated codes with applications. 1997.

[27] Steve Allpress, Carlo Luschi, and Steve Felix. Exact and approximated expressions of the log-likelihood ratio for 16-qam signals. In *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004.*, volume 1, pages 794–798. IEEE, 2004.

[28] K. R. Narayanan and G. L. Stuber. A serial concatenation approach to iterative demodulation and decoding. *IEEE Transactions on Communications*, 47(7):956–961, July 1999.

[29] Joachim Hagenauer, Elke Offer, and Lutz Papke. Iterative decoding of binary block and convolutional codes. *IEEE Transactions on information theory*, 42(2):429–445, 1996.

[30] Jing Li, K. R. Narayanan, E. Kurtas, and C. N. Georghiades. On the performance of high-rate tpc/spc codes and ldpc codes over partial response channels. *IEEE Transactions on Communications*, 50(5):723–734, May 2002.

[31] Zhenning Tao, Liang Dou, Weizhen Yan, Lei Li, Takeshi Hoshida, and Jens C. Rasmussen. Multiplier-free intrachannel nonlinearity compensating algorithm operating at symbol rate. *J. Lightwave Technol.*, 29(17):2570–2576, Sep 2011.

[32] Alireza Hosseinzade. A model for self-phase modulation noise in fiber-link communication systems. *Master thesis*, 2020.

[33] Ezra Ip and Joseph M. Kahn. Compensation of dispersion and nonlinear impairments using digital backpropagation. *J. Lightwave Technol.*, 26(20):3416–3425, Oct 2008.

[34] A. Mecozzi, C. B. Clausen, and M. Shtaif. Analysis of intrachannel nonlinear effects in highly dispersed optical pulse transmission. *IEEE Photonics Technology Letters*, 12(4):392–394, April 2000.

[35] A. Mecozzi and R. Essiambre. Nonlinear shannon limit in pseudolinear coherent systems. *Journal of Lightwave Technology*, 30(12):2011–2024, June 2012.

APPENDICES

To make sure that our gradient free iterative pre-compensation algorithm converges, we introduce two constant values:
1- Constant factor (CF): At each iteration (except iteration 1), learned C-matrix is multiplied by this factor.

$$A_i(t) = A_{i-1}(t) - CF \times A \times C_i - 1 \tag{6}$$

2- Discount Factor (DF): At each iteration, learned C-matrix is multiplied by DF to the power of the iteration number minus one.

$$A_i(t) = A_{i-1}(t) - DF^{iter-1} \times A \times C_i - 1 \tag{7}$$

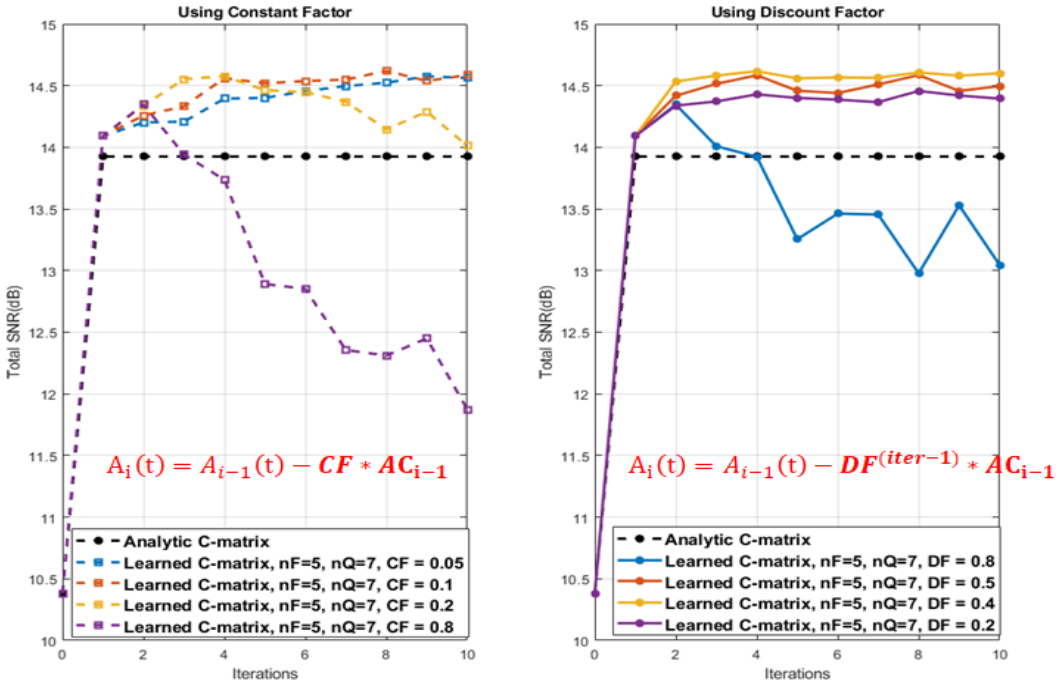


Figure 11: Convergence of the iterative pre-compensation