

**UNIVERSIDADE ESTADUAL PAULISTA**  
**"JÚLIO DE MESQUITA FILHO"**  
**CAMPUS DE SÃO JOÃO DA BOA VISTA**

**RÔMULO APARECIDO DE PAULA JUNIOR**

**Utilização de regressão logística para mitigação de não linearidades em sistemas de comunicações ópticas coerentes digitais**

São João da Boa Vista

2019

**Rômulo Aparecido de Paula Junior**

**Utilização de regressão logística para mitigação de não linearidades em sistemas de comunicações ópticas coerentes digitais**

Trabalho de Graduação apresentado ao Conselho de Curso de Graduação em Engenharia de Telecomunicações do Campus de São João da Boa Vista, Universidade Estadual Paulista, como parte dos requisitos para obtenção do diploma de Graduação em Engenharia de Telecomunicações .

Orientador: Prof<sup>o</sup> Dr. Ivan Aritz Aldaya Garde

São João da Boa Vista

2019

Paula Júnior, Rômulo Aparecido de

Utilização de regressão logística para mitigação de não linearidades em sistemas de comunicações ópticas coerentes digitais / Rômulo Aparecido de Paula Júnior -- São João da Boa Vista, 2019.  
41 p. : il. color.

Trabalho de Conclusão de Curso – Câmpus Experimental de São João da Boa Vista – Universidade Estadual Paulista “Júlio de Mesquita Filho”.

Orientador: Prof. Dr. Ivan Aritz Aldaya Garde

Bibliografia

1. Comunicações ópticas 2. Inteligência artificial 3. Telecomunicações

CDD 23. ed. – 621.382

Ficha catalográfica elaborada pela [Biblioteca-BJB](#)

Bibliotecário responsável: João Pedro Alves Cardoso – CRB-8/9717

**UNIVERSIDADE ESTADUAL PAULISTA**  
**“JÚLIO DE MESQUITA FILHO”**  
**CÂMPUS EXPERIMENTAL DE SÃO JOÃO DA BOA VISTA**  
**GRADUAÇÃO EM ENGENHARIA ELETRÔNICA E DE TELECOMUNICAÇÕES**

**TRABALHO DE CONCLUSÃO DE CURSO**

**COMPENSAÇÃO DE NÃO LINEARIDADES EM SISTEMAS DE COMUNICAÇÃO  
ÓPTICAS COERENTES USANDO REGRESSÃO LOGÍSTICA**

Aluno: Rômulo Aparecido de Paula Junior

Orientador: Prof. Dr. Ivan Aritz Aldaya Garde

Banca Examinadora:

- Ivan Aritz Aldaya Garde (Orientador)
- Leonardo de Melo João (Examinador)
- Marcelo Luís Francisco Abbade (Examinador)

A ata da defesa com as respectivas assinaturas dos membros encontra-se no prontuário do aluno (Expediente nº 23/2019)

São João da Boa Vista, 05 de dezembro de 2019

*"Há apenas uma maneira de evitar críticas:  
não faça nada,  
não diga nada,  
não seja nada."  
(Aristóteles)*

## AGRADECIMENTOS

Primeiramente à minha família que tanto me ajudou, tanto financeiramente quanto moralmente, sobretudo à minha mãe, Valéria, que praticamente sozinha conseguiu me dar o sustento necessário durante minha graduação. Mas também à minha irmã Agnes, meus tios, Luciana e Celso, e meus avós, Vilma e José, que também foram fundamentais para minha graduação. E também ao meu pai, que se orgulharia muito em me ver concluindo esta fase.

Aos professores que fizeram parte da minha educação durante a graduação, principalmente ao meu orientador Prof. Dr. Ivan Aritz Aldaya Garde, que aceitou o desafio de me orientar neste trabalho de conclusão de curso, fornecendo incentivo e apoio incondicionais, até mesmo em períodos de férias, que foram fundamentais para a conclusão deste trabalho.

Aos meus amigos de longa data, o Affonso, o Oscar, o Romão, o Henrique e o Lucas (Lukacs), que me deram grande apoio durante a minha graduação. E das amizades que encontrei na faculdade, sou grato ao Silvio, Homero, Zé Carlos, João Gabriel, Ricardo e Lucas, por toda ajuda que me deram no decorrer do curso. E agradeço também a todos aqueles que fizeram parte da minha graduação e que de certa forma me ajudaram quando precisei.

## RESUMO

No presente trabalho estudamos um método de compensação de não linearidades baseado em algoritmos de inteligência artificial para uma rede óptica passiva coerente digital de longa distancia. Simulou-se computacionalmente a transmissão de um sinal óptico modulado utilizando formato 16-QAM por meio de um enlace, cujo modelo inclui tanto efeitos lineares quanto não lineares. Os fenômenos de ruído e de atenuação são responsáveis por limitar o alcance de um sistema óptico, e o aumento da potência de transmissão não é uma solução trivial, pois isto ocasiona a excitação de não linearidades devido à interação do sinal com o meio. Os resultados mostram que, devido a rotação não linear de fase causada pelo efeito Kerr, a detecção por máxima verossimilhança apresenta um baixo desempenho. A técnica computacional implementada, denominada regressão logística, é capaz de traçar limiares de decisão complexos que se adaptam as constelações distorcidas recebidas. Ao utilizar o algoritmo demonstrou-se uma diminuição de cerca de 38% do valor ótimo da taxa de erro de símbolo em relação a detecção por máxima verossimilhança. Estes resultados indicam que, ao compensar tais efeitos, é possível melhorar o produto banda-distancia.

**PALAVRAS CHAVE: COMUNICAÇÕES ÓPTICAS; SISTEMAS COERENTES DIGITAIS; EFEITOS NÃO LINEARES E INTELIGÊNCIA ARTIFICIAL.**

## **ABSTRACT**

In this work we studied a method for compensating nonlinearities based on artificial intelligence for digital coherent long-reach passive optical networks. The transmission of an optical signal modulated utilizing 16-QAM format through a link, whose model considers both linear and nonlinear effects, was computationally simulated. In such systems, noise and attenuation limit the range. Unfortunately, increasing transmission power is not a feasible solution, because it causes the excitation of nonlinearities as a result the interaction between the signal and the medium. The results show that, due to the nonlinear phase rotation caused by the Kerr effect, the conventional detection by maximum likelihood presents low performance. The implemented computational technique, denominated logistic regression, is able to draw complex decision boundaries that fit the received distorted constellations. By applying the aforementioned algorithm, a decrease of up to 38% of the optimal value of the symbol error rate in relation with the maximum likelihood detection can be achieved. These results reveal that, by the compensation of such effects, it's possible to improve the bandwidth-length product.

**KEYWORDS: OPTICAL COMMUNICATIONS, DIGITAL COHERENT SYSTEMS, NON-LINEAR EFFECTS, AND ARTIFICIAL INTELLIGENCE.**



## LISTA DE ILUSTRAÇÕES

|           |  |    |
|-----------|--|----|
| Figura 1  | Regiões de decisão de uma modulação 16-QAM utilizando técnica de máxima verossimilhança. . . . .   | 16 |
| Figura 2  | Esquemático do algoritmo de regressão logística. . . . .   | 21 |
| Figura 3  | Função de custo para (a) $y^{(i)} = 1$ e para (b) $y^{(i)} = 0$ . . . . .  | 24 |
| Figura 4  | Classificação por regressão logística para (a) um único Atributo $x_1$ , (b) dois Atributos, $x_1$ e $x_2$ , (c) dois atributos porém com um limiar de decisão mais complexo, demonstrando <i>overfitting</i> , linha tracejada em verde, e <i>underfitting</i> , linha tracejada em cinza e (d) classificação em três classes . . . . .   | 26 |
| Figura 5  | Implementação do sistema LR-PON coerente. S/P: conversão de série para paralelo. CDA: conversor digital para analógico. DL: diodo laser. DP-MZM: moduladores <i>Mach-Zenhder</i> duplos em paralelo ( <i>dual parallel-Mach-Zenhder modulator</i> ). EDFA: amplificador de fibra dopada de érbio ( <i>erbium-doped fiber amplifier</i> ). AOV: atenuador óptico variável. FFMP: fibra óptica monomodo padrão. Atn.: atenuador. FPB: filtro passa-baixo. CAD: conversor analógico para digital. DSP: processamento digital de sinais. . . . . | 28 |
| Figura 6  | Constelações de um sinal recebido utilizando modulação 16-QAM que sofreu efeito linear de ruído e efeito não linear de SPM com potências de transmissão variando desde (a) 2 mW até (l) 13 mW, com passo de 1 mW entre os itens da figura. Ainda não se aplicou o algoritmo de classificação, apenas se utilizou a classe de cada símbolo para representação por cores. . . . .  | 33 |
| Figura 7  | Constelação de um sinal utilizando modulação 16-QAM com as regiões de decisão traçadas para um LOP de 2 mW (a) utilizando máxima verossimilhança e (b) utilizando regressão logística, e com um LOP de 13 mW (c) utilizando máxima verossimilhança e (d) utilizando regressão logística. . . . .   | 34 |
| Figura 8  | Constelação de um sinal 16-QAM com as regiões de decisão traçadas para o caso ótimo utilizando (a) máxima verossimilhança, com 5 mW de LOP e (b) regressão logística, com 6 mW de LOP. . . . .   | 35 |
| Figura 9  | Gráfico da SER em função da potência com a utilização de máxima verossimilhança e regressão logística utilizando regressão polinomial com grau de 1 a 5 com parâmetro de regularização $\lambda = 0$ . . . . .   | 35 |
| Figura 10 | Gráfico da SER em função da potência com a utilização de máxima verossimilhança e regressão logística utilizando regressão polinomial com grau de 1 a 5 com parâmetro de regularização $\lambda = 0,02$ . . . . .  | 36 |
| Figura 11 | Taxa de erro de símbolo em função do tamanho do conjunto de treinamento. . .   | 37 |

## LISTA DE ABREVIATURAS E SIGLAS

- ANN - *Artificial neural network*
- BL - *Bandwidth-length*
- DSP - *Digital signal processing*
- EDFA - *Erbium doped fiber amplifiers*
- FWM - *Four-wave mixing*
- ICA - *Independent component analysis*
- LOP - *Launch optical power*
- LR-PON - *Long-reach passive optical networks*
- MZM - *Mach-Zenhder modulators*
- OOK - *On-off keying*
- PON - *Passive optical networks*
- PSK - *Phase shift keying*
- QAM - *Quadrature amplitude modulation*
- SBS - *Stimulated Brillouin scattering*
- SDM - *Space division multiplexing*
- SER - *Symbol erro rate*
- SNR - *Signal-to-noise ratio*
- SPM - *Self-phase modulation*
- SRS - *Stimulated Raman scattering*
- SSMF - *Standard single mode fiber*
- WDM - *Wavelength-division multiplexing*
- XPM - *Cross-phase modulation*

## LISTA DE SÍMBOLOS

- $C$  - Capacidade máxima do canal
- $B$  - Largura de banda do canal
- $E_i$  - Campo elétrico resultante da  $i$ -ésima porta de saída da rede híbrida
- $E_s$  - Campo elétrico do sinal
- $E_{lo}$  - Campo elétrico do oscilador local
- $I_i$  - Foto-corrente gerada pelo  $i$ -ésimo fotodetector
- $I_I$  - Componente em fase do sinal fotodetectado
- $I_Q$  - Componente em quadratura do sinal fotodetectado
- $I$  - Sinal complexo fotodetectado
- $\theta$  - Vetor de parâmetros do algoritmo de regressão logística
- $X$  - Matriz de entradas do algoritmo de regressão logística
- $Y$  - Vetor de classes/saídas do algoritmo de regressão logística
- $g$  - Função de ativação
- $h$  - Função hipótese
- $Custo$  - Função de custo
- $\alpha$  - Taxa de aprendizado
- $\lambda$  - Termo de regularização
- $N_{val}$  - Tamanho do conjunto de validação
- $k$  - Grau de Expansão Polinomial
- $C(k)$  - Complexidade computacional em função do grau de Expansão Polinomial

## SUMÁRIO

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>INTRODUÇÃO</b> . . . . .  | <b>11</b> |
| 1.1      | Evolução histórica . . . . .   | 11        |
| 1.2      | Sistemas Ópticos Coerentes . . . . .   | 12        |
| 1.3      | Efeitos não lineares em Sistemas Ópticos Coerentes . . . . .   | 14        |
| 1.4      | Justificativa do trabalho . . . . .  | 15        |
| 1.5      | Organização do documento . . . . .   | 16        |
| <b>2</b> | <b>APRENDIZADO DE MÁQUINA E REGRESSÃO LOGÍSTICA</b> . . . . .  | <b>18</b> |
| 2.1      | Introdução à algoritmos de Aprendizado de máquina e conceitos básicos . . . . .                            | 18        |
| 2.2      | regressão logística . . . . .  | 21        |
| <b>3</b> | <b>MÉTODOS</b> . . . . .   | <b>28</b> |
| 3.1      | Sistema Óptico Coerente . . . . .  | 28        |
| 3.2      | Implementação do algoritmo de regressão logística . . . . .  | 30        |
| <b>4</b> | <b>RESULTADOS</b> . . . . .  | <b>32</b> |
| 4.1      | Avaliação do desempenho do algoritmo de regressão logística para compensação de não linearidades . . . . . | 32        |
| 4.2      | Requerimentos de sequência de treinamento . . . . .  | 37        |
| 4.3      | Avaliação de complexidade . . . . .  | 38        |
| <b>5</b> | <b>CONCLUSÕES</b> . . . . .  | <b>40</b> |
|          | <b>REFERÊNCIAS</b> . . . . .   | <b>41</b> |

# 1 INTRODUÇÃO

## 1.1 EVOLUÇÃO HISTÓRICA

Para atingir a crescente demanda de transferência de informação, é necessário o desenvolvimento de sistemas robustos o suficiente para acompanharem tais requisitos. De acordo com a Cisco Systems, estima-se que em 2017 o tráfego global através da rede IP foi de 122 Petabytes por mês, tendo a previsão de quase triplicar em 2022, com um tráfego de aproximadamente 396 Petabytes por mês (CISCO, 2017).

Sistemas por ondas luminosas, como são conhecidos os sistemas de comunicações ópticas, fazem uso de uma ou várias portadoras com frequências na ordem de centenas de THz. A operação em frequências tão elevadas indica que tais sistemas têm maior largura de banda disponível além do fato que as fibras ópticas apresentam uma baixa atenuação para tais frequências, permitindo a transmissão de sinais de banda larga a uma longa distancia.

Desde a década de 1950 já era conhecido que portadoras operando nas bandas ópticas resultariam em um maior produto BL (produto entre a banda oferecida e a distância em que o sistema opera, representando uma figura de mérito). Porém apenas em 1960 foi possível a implementação de um laser sendo utilizado como uma fonte óptica coerente. Assim foi necessário também o desenvolvimento de fibras ópticas de baixas perdas para serem utilizadas como guia de ondas (AGRAWAL, 2014).

Cada geração de comunicações por ondas luminosas apresenta impedimentos que requerem o desenvolvimento de tecnologias. Na primeira geração operava-se em um comprimento de onda de 0.8  $\mu\text{m}$  utilizando lasers de semicondutores GaAs, apresentando uma taxa de 45 Mb/s com um alcance de 10 km entre os receptores. Tanto a taxa de bits quanto o alcance foi amplamente melhorado na segunda geração, evoluindo para 1.7 Gb/s e 50 km, com o sistema operando em um comprimento de onda de 1.3  $\mu\text{m}$  e utilizando lasers de semicondutores InGaAsP.

O melhor comprimento de onda para a operação do sistema óptico foi introduzido na terceira geração, de 1.55  $\mu\text{m}$  e sendo utilizado até hoje. A inovação marcante da quarta geração foi a implementação da Multiplexação por Divisão de Comprimento de onda (*Wavelength Division Multiplexing*, WDM), possibilitando a multiplexação por comprimento de onda de vários canais em um único enlace.

Até a quarta geração são utilizadas técnicas de modulação por intensidade, formato conhecido como chaveamento *On-Off* (*On-Off Keying*, OOK), em que a informação está contida apenas na amplitude do sinal. Em tal modulação, o bit “1” é representado pela recepção de intensidade alta e o bit “0” pela recepção de intensidade baixa. Tal formato de modulação, devido a sua simplicidade, não requer um sistema coerente para demodulação, tendo então sua capacidade de detecção limitada apenas pelo fotodiodo receptor. O fotodiodo gera um sinal elétrico proporcional à potência óptica recebida e o circuito de decisão identifica os bits recebidos.

A quinta geração de comunicações por ondas luminosas apresenta formatos de modulação mais complexos, em que a informação está contida na fase da portadora óptica e, possivelmente, também na amplitude. Tais formatos já são utilizados em sistemas de micro-ondas há algum tempo. Porém no domínio óptico, há dificuldades para a compensação do ruído de fase, provindo dos lasers empregados,

e também há flutuações do estado de polarização na onda incidente. Apenas na quinta geração foi possível implementar tais formatos de modulação em sistemas ópticos (AGRAWAL, 2014).

Além de inovações como multiplexação em polarização dupla e a utilização de processamento digital de sinais (*Digital Signal Processing*, DSP), a quinta geração faz uso de formatos de modulação como o chaveamento por deslocamento de fase (*phase shift keying*, PSK), chaveamento por deslocamento de fase em quadratura (*quadrature phase shift keying*, QPSK) e modulação de amplitude em quadratura (*quadrature amplitude modulation*, QAM). Tais formatos modulam a portadora em fase e alguns modulam tanto em fase como em amplitude, como é o caso do QAM (AGRAWAL, 2014).

Formatos de modulação como os citados possibilitam aumentar significativamente o desempenho do sistema. Porém sua detecção no receptor não é tão simples como no caso em que se utiliza modulação OOK, sendo então necessário incluir sistemas de recepção coerentes (AGRAWAL, 2014). Tais sistemas possibilitam a extração da informação contida na fase do sinal óptico, em contraste com sistemas não coerentes, em que toda esta informação é perdida.

Conforme naturalmente o limite prático de desempenho da quinta geração é alcançado, a sexta geração tem o intuito de contornar suas limitações. Shannon introduziu uma relação que quantifica a capacidade máxima de um canal linear limitado por ruído, dada por:

$$C = B \cdot \log_2(1 + SNR), \quad (1.1)$$

em que  $B$  é a largura de banda do canal e  $SNR$  (*signal-to-noise ratio*) é a relação sinal ruído. Tal relação mostra que, de princípio, um canal pode ter sua capacidade indeterminadamente alta conforme se intensifica o nível de potência do sinal, assim aumentando o  $SNR$ . Entretanto em sistemas de comunicação por fibra óptica, devido a interação da onda de luz com o meio, efeitos não lineares dependentes da potência da onda são excitados. Efeitos não lineares causam distorção não linear e, conseqüentemente, diminuem a capacidade do sistema.

Em canais limitados a não linearidades, a partir de certos níveis de  $SNR$ , a eficiência espectral diminui consideravelmente. Indica-se então que as não linearidades limitam a potência óptica que pode ser transmitida no enlace, limitando o alcance do enlace devido a atenuação da onda por perdas.

Com a elevada demanda por capacidades cada vez maiores, uma das propostas da sexta geração é a multiplexação por divisão de espaço (*Space-Division Multiplexing*, SDM). O conceito de SDM é que, basicamente, cada modo ou cada núcleo de uma fibra constituída de vários núcleos transmite sequencias de bits independentemente como canais paralelos. Porém tal técnica apresenta alguns desafios como a excitação seletiva dos modos e a diafonia linear e não linear. A proposta que estudamos consiste na mitigação dos efeitos não lineares sofridos no sistema, com intuito de aumentar a capacidade do sistema. Mais especificamente, faz-se uso de processamento digital de sinais baseado em algoritmos de aprendizado de maquina, como regressão logística, que será o foco do trabalho.

## 1.2 SISTEMAS ÓPTICOS COERENTES

Sistemas ópticos coerentes possibilitam que informações contidas na fase do sinal sejam recuperadas pelo receptor. Para isso, o sinal recebido é combinado com a onda de um oscilador local,

mantido em uma frequência fixa, que pode ter o mesmo valor que o sinal: detecção homódina, ou um valor bem diferente do sinal: detecção heteródina. Entretanto nos interessa um terceiro tipo de detecção, pois sua implementação é mais simples, que se trata da detecção intradina. Em tal detecção, as frequências centrais do sinal e do oscilador local não são exatamente as mesmas. As diferenças de frequências descritas podem ser compensadas mediante o processamento digital de sinais. Tais sistemas são denominados de sistemas ópticos coerentes digitais.

Os primeiros sistemas coerentes digitais foram implementado em redes de longa distancia. Porem, com a redução no custo da implementação de Processamento Digital de Sinais (DSP, *Digital Signal Processing*) e a possibilidade de utilização de lasers com maior largura de linha, sistemas coerentes foram propostos para enlaces progressivamente mais curtos. Atualmente está sendo proposta a utilização de sistemas coerentes para a Rede Óptica Passiva (*Passive Optical Network*, PON), com o intuito de incrementar a taxa de 100 Gb/s para 1 Tb/s, em conjunto com a implementação de algoritmos de DSP (SUZUKI et al., 2018).

Para combinar o sinal com a onda do laser local é utilizada uma rede híbrida 2x4 (duas entradas e quatro saídas) com defasagem de 90° entre as portas. Tal configuração possibilita a obtenção das componentes em fase e em quadratura do sinal. O dispositivo é um conjunto de acopladores e defasadores que permite separar as combinações do sinal com o laser sem deslocamento de fase e com um deslocamento de 180°. As combinações do sinal com as réplicas defasadas do oscilador local são fotodetectadas logo após a saída de cada porta da rede híbrida, sendo então a detecção feita no domínio elétrico (KAMINOW; LI; WILLNER, 2008).

Dado que  $E_s$  seja o campo elétrico do sinal,  $E_{10}$  o campo elétrico do oscilador local e  $E_i$  o campo elétrico da saída da  $i$ -ésima porta, com  $i$  tomando valores de 1 a 4. Desconsiderando-se o ruído de fase do oscilador local por simplicidade, temos que o campo elétrico na saída das portas é dado por:

$$E_1 = \frac{1}{\sqrt{2}}[E_s + E_{10}], \quad (1.2)$$

$$E_2 = \frac{1}{\sqrt{2}}[E_s - E_{10}], \quad (1.3)$$

$$E_3 = \frac{1}{\sqrt{2}}[E_s + jE_{10}], \quad (1.4)$$

$$E_4 = \frac{1}{\sqrt{2}}[E_s - jE_{10}]. \quad (1.5)$$

Por conveniência, desconsidera-se os ruídos introduzidos durante a foto-detecção, como o ruído térmico e o ruído *shot*. Sendo a responsividade do fotodetector unitária, as correntes geradas pelo  $i$ -ésimo fotodetector são dadas por:

$$I_1 = \frac{1}{2}[|E_s|^2 + |E_{10}|^2 + 2\Re(E_s E_{10}^*)], \quad (1.6)$$

$$I_2 = \frac{1}{2}[|E_s|^2 + |E_{10}|^2 - 2\Re(E_s E_{10}^*)], \quad (1.7)$$

$$I_3 = \frac{1}{2}[|E_s|^2 + |E_{10}|^2 + 2\Im(E_s E_{10}^*)], \quad (1.8)$$

$$I_4 = \frac{1}{2}[|E_s|^2 + |E_{10}|^2 - 2\Im(E_s E_{10}^*)], \quad (1.9)$$

Em que o asterisco indica a operação de complexo conjugado. Sendo assim, podemos derivar a componente em fase do sinal:

$$I_I = I_1 - I_2 = 2\Re(E_s E_{10}^*). \quad (1.10)$$

Da mesma forma, temos que a componente em quadratura do sinal é dado por:

$$I_Q = I_3 - I_4 = 2\Im(E_s E_{10}^*). \quad (1.11)$$

Finalmente, temos que o sinal complexo detectado é dado por:

$$I = I_I + jI_Q = 2\Re(E_s E_{10}^*) + j2\Im(E_s E_{10}^*) = 2E_s E_{10}^*. \quad (1.12)$$

### 1.3 EFEITOS NÃO LINEARES EM SISTEMAS ÓPTICOS COERENTES

Dentre os efeitos não lineares devido ao espalhamento da luz que interage com o meio, temos o Espalhamento Estimulado de Brillouin (*Stimulated Brillouin Scattering*, SBS) e o Espalhamento Estimulado de Raman (*Stimulated Raman Scattering*, SRS). A excitação destes efeitos depende principalmente da intensidade da onda que se propaga no meio assim como sua largura de linha. Em particular, a excitação do efeito SBS necessita de uma largura de linha estreita o suficiente para que os fônons gerados pelo efeito estejam em coerência de fase (GARMIRE, 2017). Tais efeitos não afetam nosso sistema pois em sistemas ópticos coerentes não há portadora, também conhecida como portadora suprimida. Por conseguinte, a largura de linha em que o sistema em estudo opera é alta o suficiente para não excitar o efeito SBS. O efeito SRS possui um limiar de intensidade muito alto para ser excitado, sendo então o sistema em estudo incapaz de estimulá-lo.

A respeito dos efeitos não lineares que ocorrem devido a alterações do índice de refração, o efeito Kerr se origina de uma resposta de polarização não linear das partículas do meio sob influência de uma luz de intensidade relativamente alta. Este efeito pode ser descrito como flutuações do valor do índice de refração do meio. Assim como o efeito Pockels, geralmente utilizado para modular a fase do sinal em moduladores do tipo *Mach-Zehnder* (*Mach-Zehnder Modulator*, MZM), o efeito Kerr causa uma modulação em fase da onda que é resultante da variação do índice de refração. Porém a variação é dependente da amplitude do campo elétrico no efeito Pockels, e no efeito Kerr depende da intensidade, a qual é proporcional ao quadrado da amplitude do campo elétrico (KEISER, 2010).

O efeito Kerr pode causar auto-modulação de fase (*Self Phase Modulation*, SPM), modulação cruzada de fase (*Cross Phase Modulation*, XPM) e mistura de quatro ondas (*Four Wave Mixing*, FWM). XPM e FWM são ocasionados apenas quando há mais de uma onda propagando pelo meio, como em sistemas WDM (KEISER, 2010). O único efeito que nos interessa é a SPM, pois a proposta deste trabalho é a simulação da transmissão de apenas um canal sendo propagado no enlace óptico, não



existindo então efeitos como XPM e FWM.

SPM ocorre quando um pulso modifica sua própria fase. O efeito da SPM causa mudanças no espectro de frequência e uma rotação dependente da potência instantânea dos símbolos no diagrama de constelação do sinal. Sendo então o efeito SPM o maior impedimento para sistemas ópticos coerentes com apenas um canal, o foco do trabalho se dará na compensação de tal efeito.

#### 1.4 JUSTIFICATIVA DO TRABALHO

Como discutido na seção anterior, as alterações de fase indesejadas causadas pelo efeito Kerr é o maior impedimento não-linear para sistemas ópticos coerentes. Para aumentar a capacidade de tais sistemas, é justificável a implementação de algoritmos de aprendizado de máquina para mitigar os efeitos provenientes da SPM.

Um símbolo em uma modulação digital indica uma sequência de *bits* representada por um ponto no plano complexo. A quantidade de *bits* pertencentes a um símbolo é dependente do formato de modulação. Um formato de modulação que possui uma maior quantidade de *bits* por símbolo apresenta uma maior eficiência espectral. Apesar da maior eficiência de transmissão, tem-se um alfabeto maior de símbolos, aumentando a proximidade entre estes. Com uma menor distância entre os símbolos, se aumenta a probabilidade de erro na detecção. Ao se utilizar o formato de 16-QAM, em relação ao formato de QPSK, é capaz de representarmos o dobro de *bits* por símbolo, porém apresentando uma maior taxa de erros.

Máxima verossimilhança é uma técnica estatística utilizada para traçar as regiões de decisão do receptor. No caso, as regiões são fixas e a decisão é feita considerando a mínima distância entre o símbolo recebido e os símbolos ideais pertencentes ao formato de modulação digital (LATHI; DING, 2012). A Figura 1 ilustra as regiões de decisão para um formato de modulação 16-QAM obtidas a partir do método de máxima verossimilhança. Cada área que se difere pela cor indica uma diferente região de decisão. Os pontos representados pelos marcadores em cruz indicam os símbolos ideais normalizados em relação aos símbolos de maior potência.

O efeito Kerr causa uma modulação em fase, a qual gera uma rotação dos símbolos recebidos em um diagrama de constelação. Tais alterações na fase do sinal resulta em símbolos invadindo as regiões de decisão vizinhas. É intuitivo que tal fenômeno eleva a probabilidade de erros de símbolos no sistema.

A utilização de técnicas de aprendizado de máquina já vem sendo utilizada para compensação de não linearidades em sistemas de comunicações ópticas. Há trabalhos em que demonstra o desempenho de algoritmos como Redes Neurais Artificiais (*Artificial Neural Networks*, ANN) (FUKUMOTO et al., 2018) e a utilização de algoritmos de clusterização (capazes de agrupar dados conforme seu nível de semelhança) (TORRES et al., 2015) para tal finalidade. Porém, até onde sabemos, nunca se utilizou o algoritmo de regressão logística com este intuito.

O algoritmo de regressão logística apresenta a vantagem de possuir baixo custo computacional e de possuir implementação relativamente pouco complexa. O desempenho da regressão logística será avaliado em função da Taxa de Erro de Símbolo (*Symbol Error Rate*, SER). O funcionamento do algoritmo será discutido mais detalhadamente no Capítulo 2.

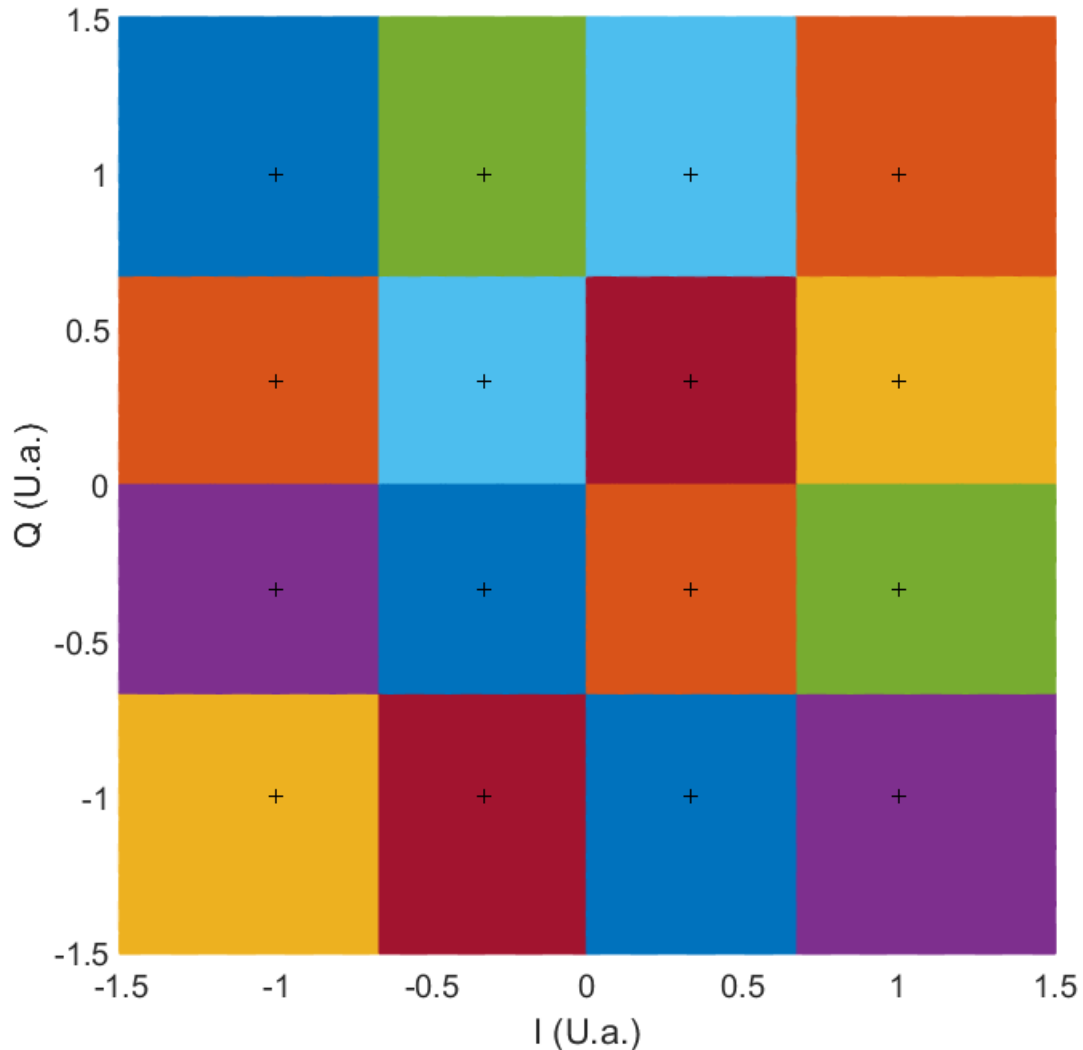


Figura 1 – Regiões de decisão de uma modulação 16-QAM utilizando técnica de máxima verossimilhança.

## 1.5 ORGANIZAÇÃO DO DOCUMENTO

Este trabalho tem como objetivo desenvolver e avaliar a utilização do algoritmo de aprendizado de máquina denominado regressão logística para a mitigação de não linearidades em um sistema de comunicação coerente. O trabalho é dividido como segue:

O **Capítulo 2** apresenta os conceitos básicos da área de aprendizado de máquina, com o propósito do leitor se familiarizar com o funcionamento do algoritmo de regressão logística. É desenvolvido então o equacionamento que rege a classificação por regressão logística, assim como suas vantagens e limitações.

O **Capítulo 3** apresenta a metodologia a ser empregada no trabalho, assim como suas ferramentas e métodos computacionais. O conteúdo do **Capítulo 3** se baseia na implementação do sistema óptico no *software VPI Transmission Maker* e na implementação de regressão logística no *software Matlab*.

O **Capítulo 4** apresenta uma comparação da taxa de erro de símbolo sofrida pelo sistema utilizando detecção por máxima verossimilhança e por regressão logística. O **Capítulo 4** demonstra também a

configuração do algoritmo em que se obteve o melhor desempenho, além de apresentar uma análise do efeito do tamanho do seu conjunto de treinamento e de sua complexidade computacional.

O **Capítulo 5** apresenta uma conclusão do trabalho.

## 2 APRENDIZADO DE MÁQUINA E REGRESSÃO LOGÍSTICA

### 2.1 INTRODUÇÃO ÀS ALGORITMOS DE APRENDIZADO DE MÁQUINA E CONCEITOS BÁSICOS

Aprendizado de máquina se refere a algoritmos baseados em estatística, que são capazes de reconhecer padrões em distribuições de dados. O aprendizado se dá durante um processo que chama-se de treinamento, em que geralmente, o algoritmo realiza iterações para otimizar os parâmetros do modelo. A finalidade de tal otimização é a de modelar os padrões existentes em um conjunto de dados, chamados de exemplos de treinamento.

Apesar de existir trabalhos a respeito de tal área em um artigo que data 1959, que descreve o aprendizado de um computador em jogar xadrez (SAMUEL, 1959), apenas atualmente tais algoritmos estão se tornando uma tendência mundial. A popularidade atual é devida ao avanço tecnológico, como o da área da eletrônica e de telecomunicações, e mais especificamente, da computação distribuída. Pode-se citar vários exemplos do uso de algoritmos de aprendizado de máquina no cotidiano, como é o caso de: mecanismos de buscas achar os melhores resultados, assim como anúncios direcionados para um usuário individualmente, filtros *anti-spam* utilizados em *e-mails*, reconhecimento de rostos em imagens e reconhecimento de voz pelo celular (SHALEV-SHWARTZ; BEN-DAVID, 2014).

É altamente vantajoso utilizar tais algoritmos em aplicações em que a complexidade do problema é alta, impossibilitando uma pessoa de programar explicitamente todas as instruções a serem seguidas (SHALEV-SHWARTZ; BEN-DAVID, 2014). O fato de não serem programados para realizarem tarefas explicitamente, algoritmos de aprendizado de máquina se adaptam e aprendem com a experiência, apresentando mudanças de comportamento se houver alteração na estrutura dos dados a serem utilizados para treinamento. Como a área de aprendizagem é bastante variada, justifica-se a classificação de algoritmos em relação ao tipo do aprendizado, separando-os em categorias. Considerando a relação da entrada com a saída podemos classificá-los como:

- **Aprendizado supervisionado:** No aprendizado supervisionado os dados introduzidos no algoritmo são rotulados, ou seja, as classes são conhecidas nos exemplos de treinamento. Estes algoritmos podem ser:
  1. **De regressão:** Possibilita o mapeamento das variáveis de entrada em uma função contínua, como a previsão do preço de uma casa em função de seu tamanho. Sendo assim a saída pode assumir um valor contínuo dentro dos limites estabelecidos pela modelagem do problema. Um exemplo simples e muito utilizado de algoritmo de regressão é a Regressão Linear, capaz de aproximar uma reta ótima que descreve uma distribuição de dados (ALPAYDIN, 2010) e também de regressão por intermédio de Redes Neurais Artificiais.
  2. **De classificação:** Resulta em modelos que preveem os resultados representados por meio de uma saída discreta. Um algoritmo de classificação aprende por meio dos dados de entrada e é capaz de prever a qual classe novos dados de entrada pertencem. O problema de

classificação pode consistir de apenas duas classes, de classificação binária ou de múltiplas classes, sendo o último uma generalização da classificação binária. Alguns exemplos de algoritmos de classificação bem conhecidos são: Máquina de vetores de suporte (*Support vector machines*), regressão logística e por ANNs, que podem ser utilizadas tanto para regressão como para classificação. (ALPAYDIN, 2010).

- **Aprendizado Não Supervisionado:** Caso em que não são definidas as saídas dos dados de treinamento, ou seja, eles não são rotulados. Tais algoritmos são utilizados em problemas em que a estrutura dos dados de treinamento não possuem classes, de forma que o resultado se dará de forma a agrupar os dados conforme suas similaridades (SHALEV-SHWARTZ; BEN-DAVID, 2014).
  1. **De clusterização:** Capaz de agrupar dados em grupos, conhecidos como *clusters*, de acordo com as similaridades de suas características (ALPAYDIN, 2010). Também apresentariam uma boa alternativa para a compensação de não linearidades em sistemas ópticos coerentes digitais. Pode-se nomear alguns exemplos como *K-means* e *Gaussian Mixture Model*.
  2. **De não clusterização:** Permite estruturar um padrão a partir de um ambiente em que os dados não são bem definidos, como é o caso do algoritmo de análise de componentes independentes (*Independent Component Analysis*, ICA), que possibilita extrair dados individuais de um conjunto. Tais algoritmos são utilizados em problemas como *Cocktail Party*, possibilitando a identificação de vozes individuais em uma situação em que elas estão misturadas (THARWAT, 2018). Outra aplicação útil apresentada por tais algoritmos é a redução de dimensionalidade, que nos permite definir um conjunto de dados utilizando um espaço com menos dimensões.
- **Aprendizado Semi-Supervisionado:** Combinação de Aprendizado Supervisionado e de Aprendizado Não Supervisionado, utiliza ambos dados rotulados e não rotulados (ALPAYDIN, 2010).
- **Aprendizado Reforçado ou por Reforço:** Não há conjunto de treinamento, o sistema escolhe ações e recebe respostas do ambiente, utilizadas como indicador qualitativo de desempenho (ALPAYDIN, 2010).

Além da capacidade de se modelar soluções que se enquadram bem para um determinado conjunto de treinamento, o mais importante é a capacidade de generalizar a representação para dados futuros. Têm-se o intuito de prever corretamente resultados para dados que não pertencem ao conjunto de treinamento (ALPAYDIN, 2010). É apropriado então apresentar métricas que nos dizem a confiabilidade dos resultados dos algoritmos de aprendizado de máquina.

É conhecido como *underfitting* a condição em que o modelo não é capaz de representar bem o conjunto de dados de treinamento e portanto se generalizar para outros dados. O *underfitting* pode ocorrer por uma complexidade baixa demais do modelo ou por poucas iterações de treinamento. A consequência do *underfitting* é que o modelo não é capaz de caracterizar um dado problema (SHALEV-SHWARTZ; BEN-DAVID, 2014).

Por outro lado, o termo *overfitting* é o nome dado para situações em que um modelo representa muito bem o conjunto de treinamento, porém, ao utilizar o modelo para representar outros dados, não se obtém um bom desempenho. Ocorre quando o modelo é treinado por muitas iterações e aprende imprecisões e padrões aleatórios presentes nos dados de treinamento (SHALEV-SHWARTZ; BEN-DAVID, 2014). Um modo de contornar tal impedimento é a utilização de uma técnica chamada de regularização, na qual se penalizam alguns parâmetros do modelo a fim de evitar o *overfitting* (SHALEV-SHWARTZ; BEN-DAVID, 2014).

Outra técnica utilizada para evitar o *overfitting* é dividir os dados em seções, em que uma é utilizada para o treinamento e a outra é chamada de conjunto de validação, que é utilizada para avaliar o desempenho do sistema a cada iteração. A validação é uma avaliação realizada, a cada iteração, em um conjunto de dados reservado. Como tais dados não são utilizados para o treinamento, ao ocorrer *overfitting* serão obtidos erros ao aplicar o algoritmo no conjunto de validação. É comum também dividir outra seção de dados que servirá de teste após o modelo passar pelo treinamento, tal conjunto é chamado de conjunto de teste (ALPAYDIN, 2010).

Entretanto a separação de dados entre treinamento, validação e teste não é trivial. Dentre os métodos de separação de dados, podemos citar:

- **Método *Holdout*:** No método *Holdout* uma seção fixa dos dados são separados para validação, enquanto o restante é utilizado para treinamento. Deve-se atentar em manter um balanceamento de classes, para evitar a distorção dos resultados (YADAV; SHUKLA, 2016).
- **Método *Holdout Repetido*:** Uma variação do método *Holdout*, em que a cada iteração os dados são divididos entre as seções de treinamento e de validação aleatoriamente de acordo com proporções pré-estabelecidas. Uma desvantagem é que alguns exemplos podem ser selecionados mais ou menos do que outros para os conjuntos de treinamento e de validação, o que pode distorcer os resultados (KIM, 2009).
- **Método *K-Fold*:** No método *K-Fold* os dados são divididos em um número pré-estabelecido de partições. A cada iteração se usa uma destas partições para validação, enquanto o restante dos dados são utilizados para treinamento. Tal método tem a vantagem de se utilizar todos exemplos para validação apenas uma vez (ALPAYDIN, 2010).
- **Método *Leave-One-Out*:** Um caso particular do método *K-Fold* em que, a cada iteração, apenas um exemplo é utilizado para validação e o restante para treinamento. No caso do método *Leave-One-Out*, o número de iterações é a mesma que o tamanho do conjunto de dados de treinamento (ALPAYDIN, 2010).

Tais métodos de separação de dados apresentam a vantagem de avaliar o modelo a cada iteração, utilizando dados que não estão inclusos no conjunto de treinamento. Avaliando a rede apenas em relação aos dados de treinamento, pode-se levar a obtenção de resultados tendenciosos, o que pode causar *overfitting*.

## 2.2 REGRESSÃO LOGÍSTICA

Regressão logística é um algoritmo de aprendizado de máquina de classificação. A partir de um conjunto de observação, o modelo é capaz de aprender e realizar previsões, indicando a qual classe melhor se encaixa novos dados de entrada (HOSMER; LEMESHOW, 2000). Sua grande vantagem é sua implementação relativamente simples em comparação a outros algoritmos de classificação e demonstra eficiência até mesmo para alguns problemas complexos.

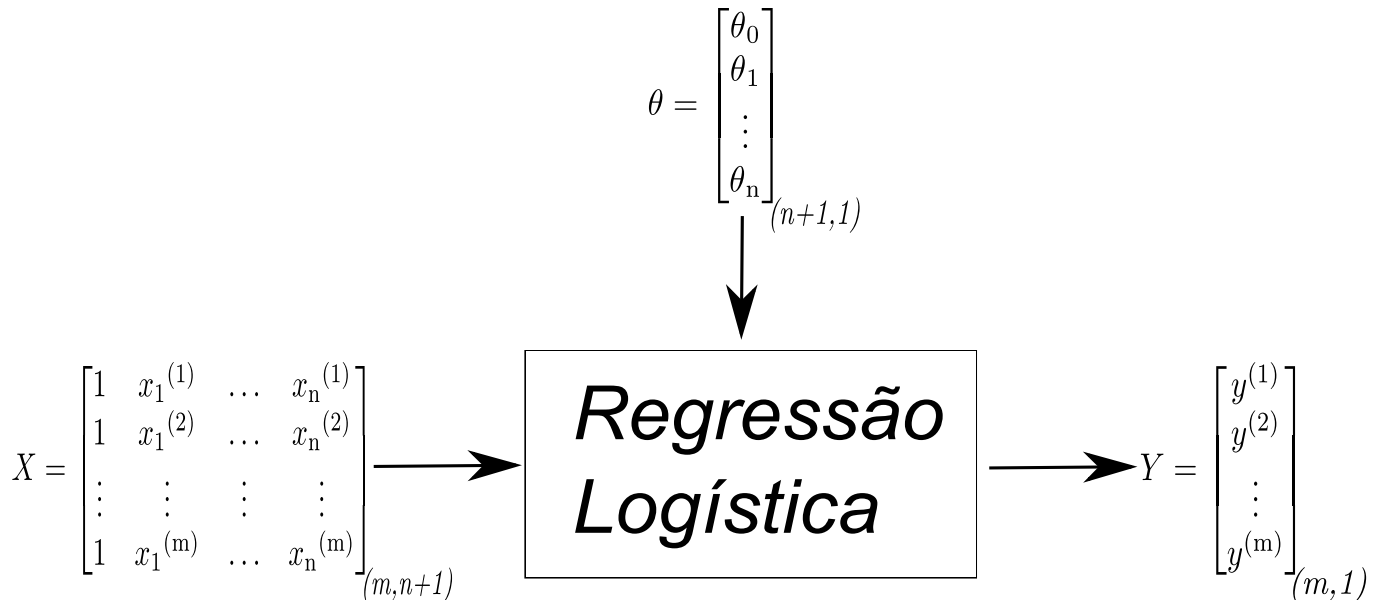


Figura 2 – Esquemático do algoritmo de regressão logística.

Denominamos como  $X$  a matriz de entradas e  $Y$  o vetor de saídas, tais saídas podendo ser chamadas de rótulos ou classes ao considerar um algoritmo de classificação. O número de exemplos que pertencem ao treinamento é dado por  $m$  e a quantidade de atributos (*features*) por exemplo de treino é dado por  $n$ .

A matriz  $X$  é definida como:

$$X = \begin{bmatrix} x_0^{(1)} & x_1^{(1)} & \dots & x_n^{(1)} \\ x_0^{(2)} & x_1^{(2)} & \dots & x_n^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ x_0^{(m)} & x_1^{(m)} & \dots & x_n^{(m)} \end{bmatrix}. \quad (2.1)$$

Um elemento da matriz de entradas  $X$  é referenciado por  $x_j^{(i)}$ , no qual o subscrito  $j$  indica o  $j$ -ésimo Atributo de uma entrada, enquanto o sobrescrito  $i$  indica o  $i$ -ésimo exemplo de treinamento. Tem-se que  $X$  possui dimensões de  $m$  linhas por  $(n + 1)$  colunas, leva-se em consideração o atributo de *bias*  $x_0^{(i)}$ . O termo de *bias* é multiplicado por um parâmetro capaz de realizar alterações na função de hipótese para melhor adequação ao problema (HOSMER; LEMESHOW, 2000). Por convenção, o valor do *bias*  $x_0^{(i)}$  será sempre igual a  $1 \forall i \in [1, m]$ .

Os vetores que representam a saída  $Y$  e os parâmetros  $\theta$  são dados pelas seguintes equações:

$$Y = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}, \quad (2.2)$$

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}. \quad (2.3)$$

No caso de um problema simples, em que temos apenas um atributo de entrada, temos que  $n = 1$ . Devido ao termo de *bias*,  $X$  tem dimensões de  $m$  linhas por 2 colunas. É trivial que, quanto mais atributos temos para descrever nossos exemplos, mais complexo será o modelo desenvolvido pelo algoritmo. Também é fato que, até certo ponto, quanto mais exemplos de treinamento, melhor se dará a modelagem.

Por ser um algoritmo de classificação, na regressão logística, em sua versão mais simples, seu vetor de saídas só pode tomar os valores binários de 0, que indica negativo, e 1, que indica positivo (HOSMER; LEMESHOW, 2000). Como estamos lidando com um problema que faz uso de 16 classes, têm-se então dois métodos de classificação, dados por:

- Um contra todos (*One-versus-All*): Consiste em classificar uma classe por vez. No caso a classe que está sendo classificada tem a saída representada por 1 (positiva), e todas outras são representadas por 0 (negativa) (SHALEV-SHWARTZ; BEN-DAVID, 2014).
- *One-versus-One* baseado na estrutura de árvores de decisão: Esta abordagem divide o problema de classificação em partes menores, mapeando as classes levando em consideração as classificações feitas em cada etapa (SHALEV-SHWARTZ; BEN-DAVID, 2014). Por exemplo, no caso de dezesseis classes, dividimos o problema em quatro classificações, e é possível identificar cada uma das classes através da combinação dos resultados obtidos nas quatro classificações anteriores.

A função hipótese  $h$  é a função responsável por mapear  $X$  em  $Y$  ( $h : X \rightarrow Y$ ), e, para um algoritmo de regressão logística, é dada por:

$$h(x^{(i)}) = h_{\theta}(x^{(i)}) = g(\theta_0 x_0^{(i)} + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} + \dots + \theta_n x_n^{(i)}) = g(x^{(i)}\theta). \quad (2.4)$$

Na equação 2.4 a função  $g$  é chamada de função de ativação. Tal função é responsável por definir a probabilidade de se obter uma saída positiva. Tal probabilidade é definida levando em consideração o vetor  $x^{(i)}$ , que representa os atributos do  $i$ -ésimo exemplo de treinamento, ponderados pelos parâmetros



$\theta$  (NG, 2012). A função  $g$  a ser utilizada é uma função sigmóide, dada por:

$$g(z) = \frac{1}{1 - e^{-z}}, \quad (2.5)$$

de forma que podemos reescrever a função de hipótese como:

$$h(x^{(i)}) = \frac{1}{1 - e^{-x^{(i)}\theta}}. \quad (2.6)$$

A função de hipótese pode ser interpretada como a probabilidade da saída  $y^{(i)}$  ser igual a 1, ou seja, a probabilidade de  $y^{(i)}$  pertencer a classe em questão, que pode ser escrita formalmente como:

$$h(x^{(i)}) = P(y = 1/x; \theta). \quad (2.7)$$

A saída  $y^{(i)}$  pode ser escrita como:

$$y^{(i)} = \begin{cases} 1, & h(x^{(i)}) \geq 0,5, \quad x^{(i)}\theta \geq 0, \\ 0, & h(x^{(i)}) < 0,5, \quad x^{(i)}\theta < 0. \end{cases} \quad (2.8)$$

De princípio, deve-se inicializar os parâmetros  $\theta$  como zero. Tais parâmetros serão otimizados por iterações, de forma a encontrarem valores ótimos para o mapeamento das entradas em relação às saídas. Tem-se o propósito de encontrar valores de  $\theta$  de forma que  $h(x^{(i)})$  seja próximo a  $y^{(i)}$  nos exemplos de treinamento (NG, 2012).

Para achar os valores ótimos de  $\theta$ , é requerida uma métrica que indique o grau de semelhança entre a saída do sistema e os dados de treinamento. Esta métrica é chamada de função de custo e, para o caso de regressão logística, é dada por:

$$Custo[h(x^{(i)}), y^{(i)}] = \begin{cases} -\log[h(x^{(i)})], & y^{(i)} = 1, \\ -\log[1 - h(x^{(i)})], & y^{(i)} = 0. \end{cases} \quad (2.9)$$

Tal abordagem é chamada de Perda de Entropia Cruzada (*Cross Entropy Loss*) (SHALEV-SHWARTZ; BEN-DAVID, 2014), podendo ser reformulada de forma compacta como:

$$Custo[h(x^{(i)}), y^{(i)}] = -y^{(i)} \cdot \log[h(x^{(i)})] - (1 - y^{(i)}) \cdot \log[1 - h(x^{(i)})]. \quad (2.10)$$

As Figura 3(a) e (b) mostram a função custo utilizada para, respectivamente,  $y^{(i)} = 1$  e  $y^{(i)} = 0$ . É possível observar que a função custo tende a um número alto se  $h(x^{(i)})$  tem valores muito diferentes do  $y^{(i)}$  em questão. Se a saída do exemplo de treinamento  $y^{(i)}$  é zero e  $h(x^{(i)})$  é aproximadamente um, teremos como resultado então um custo elevado. Podemos formalizar isto matematicamente através das seguintes equações:

$$\lim_{h(x^{(i)}) \rightarrow 0} Custo[h(x^{(i)}), y^{(i)}] = \infty \mid y^{(i)} = 1, \quad (2.11)$$

$$\lim_{h(x^{(i)}) \rightarrow 1} Custo[h(x^{(i)}), y^{(i)}] = \infty \mid y^{(i)} = 0. \quad (2.12)$$

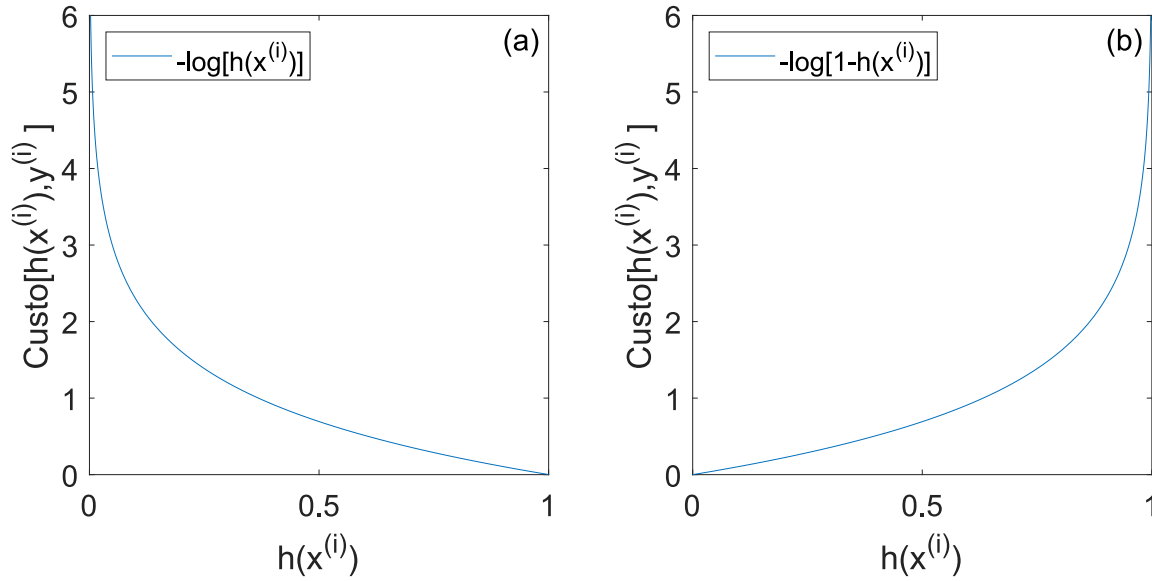


Figura 3 – Função de custo para (a)  $y^{(i)} = 1$  e para (b)  $y^{(i)} = 0$ .

O valor obtido na função custo mensura, de certa forma, o quão distante as hipóteses obtidas estão dos valores de saída dos exemplos de treinamento, penalizando previsões erradas. Nos interessa uma média da função custo para todo o conjunto de treinamento, chamada de  $J$ . Esta função de custo que considera os  $m$  exemplos de treinamento é formulada por:

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m \text{Custo}[h(x^{(i)}), y^{(i)}] = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \cdot \log[h(x^{(i)})] + (1 - y^{(i)}) \cdot \log[1 - h(x^{(i)})]]. \quad (2.13)$$

Tem-se o objetivo de encontrar valores de  $\theta$  de forma a minimizar a função custo  $J$ . O algoritmo tradicionalmente utilizado para este fim é chamado de *Gradient Descent*, o qual produz alterações nos valores de  $\theta$  de forma a se "deslocar" em direção ao mínimo da função  $J$  (NG, 2012). Tal algoritmo pode ser descrito como:

$$\theta_j' = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) = \theta_j - \frac{\alpha}{m} \sum_{i=1}^m [h(x^{(i)}) - y^{(i)}] x_j^{(i)}, \quad \forall j \in [0, n]. \quad (2.14)$$

Na equação 2.14 o termo  $\theta_j'$  representa o novo valor que  $\theta_j$  deve tomar. É necessário atualizar simultaneamente o valor de cada  $\theta_j$  pertencente ao vetor  $\theta$ . A cada iteração se atualiza todos elementos de  $\theta$ , antes da próxima iteração, pois senão os valores podem não convergir ao mínimo global. É válido ressaltar que essa versão do algoritmo é chamada de *Batch Gradient Descent*, pois usa todos exemplos de treinamento na atualização de cada parametro de  $\theta$ , evidenciado pela somatória da equação 2.14.

O termo  $\alpha$  na equação 2.14 é conhecido como Taxa de Aprendizado (*Learning Rate*).  $\alpha$  controla a quantidade de variação do valor de  $\theta$  a cada iteração. Deve-se escolher o valor de  $\alpha$  de forma adequada pois, se for muito baixo, levará muitas iterações para os valores de  $\theta$  convergirem. Se  $\alpha$  tiver um valor muito alto, o valor de  $J(\theta)$  pode aumentar e os valores de  $\theta$  não irão convergir.

É necessário, para casos em que não temos Atributos de entrada o suficiente, utilizarmos de técnicas

para se ampliar o nível de complexidade que a função hipótese é capaz de descrever. Para traçar curvas de limiares de decisão mais apropriadas, foi utilizada uma técnica chamada de mapeamento polinomial de atributos. Tal técnica é capaz de modelar os atributos dos exemplos de treinamento em forma de um polinômio de grau qualquer, expandindo sua dimensionalidade. A utilidade de tal técnica é aumentar a não linearidade das curvas que devem ajustar uma classe, traçando regiões de decisão mais complexas.

No trabalho, utilizamos uma permutação dos atributos de um exemplo de treinamento para nos ajustarmos aos padrões não lineares dos dados. Temos que  $x_1^{(i)}$  representa a amplitude no eixo real do símbolo e  $x_2^{(i)}$  representa a amplitude no eixo imaginário do símbolo. Para obtermos então um polinômio de grau  $k$  utilizamos a expressão:

$$\begin{aligned} x^{(i)} &= 1 + (x^{(i)})_k \\ &= 1 + (x_1^{(i)})^k + (x_1^{(i)})^{k-1}(x_2^{(i)}) + (x_1^{(i)})^{k-2}(x_2^{(i)})^2 + \dots \\ &\quad + (x_1^{(i)})^2(x_2^{(i)})^{k-2} + (x_1^{(i)})(x_2^{(i)})^{k-1} + (x_2^{(i)})^k + (x^{(i)})_{k-1}. \end{aligned} \quad (2.15)$$

Na equação 2.15, o termo  $(x^{(i)})_k$  representa os atributos de treinamento do  $i$ -ésimo exemplo de treinamento depois de se utilizar a técnica de mapeamento polinomial de atributos com um grau  $k$ . O termo  $(x^{(i)})_{k-1}$  é o exemplo de treinamento utilizando da técnica de mapeamento polinomial de atributos com um grau  $k - 1$ . A técnica consiste em utilizar os Atributos dados para representarmos os exemplos de treinamento em uma dimensão maior. É útil em problemas em que não se tem muitos Atributos por exemplo de treinamento.

Aumentando a complexidade das curvas que classificam os exemplos de treinamento, é natural se deparar com o problema de *overfitting*. Podemos contornar o problema de *overfitting* utilizando a técnica de regularização, que consiste em atenuar os valores dos parâmetros  $\theta$  sem a necessidade de reduzir a quantidade de Atributos.

Considerando a regularização, podemos aprimorar a equação 2.13 a escrevendo como:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log(h(x^{(i)})) + (1 - y^{(i)}) \log(1 - h(x^{(i)}))] + \frac{\lambda}{2m} \sum_{j=1}^n (\theta_j)^2. \quad (2.16)$$

Na equação 2.16 o termo  $\lambda$  é chamado de termo de regularização. O valor de  $\lambda$  deve ser escolhido de acordo com as necessidades do problema e é responsável por penalizar os termos  $\theta_j$  para  $j \neq 0$ . Podemos perceber que, ao escolher  $\lambda$  igual a zero, a equação 2.16 converge à equação 2.13. As atualizações causadas em  $\theta$  durante o *Gradient Descent* então podem ser re-escritas como:

$$\begin{cases} \theta_0' = \theta_0 - \frac{\alpha}{m} \sum_{i=1}^m [h(x^{(i)}) - y^{(i)}] x_0^{(i)} \\ \theta_j' = \theta_j (1 - \alpha \frac{\lambda}{m}) - \frac{\alpha}{m} \sum_{i=1}^m [h(x^{(i)}) - y^{(i)}] x_j^{(i)}, \forall j \in [1, n]. \end{cases} \quad (2.17)$$

A equação 2.16 e a equação 2.17 deixam claro que o parâmetro  $\theta_0$  não é penalizado pela regularização. Tal decisão é tomada por convenção, apesar de que, na prática, os resultados não são tão diferentes penalizando  $\theta_0$  ou não (NG, 2012). Na equação 2.16 utilizamos o termo de somatório em relação a  $(\theta_j)^2$  do que em relação à  $\theta_j$  pois, se houver valores de  $\theta_j$ s equivalentes porém com sinais

trocados, ambos se anulariam, o que introduziria erros na penalização. Deve-se levar em consideração que, um valor alto de  $\lambda$  pode causar consequências como *underfitting*. O ato de minimizar a função custo da equação 2.16 utilizando um  $\lambda$  elevado reduziria demasiadamente os valores de  $\theta_j$ , resultando em um modelo simples e inadequado para tratar devidamente o problema de classificação.

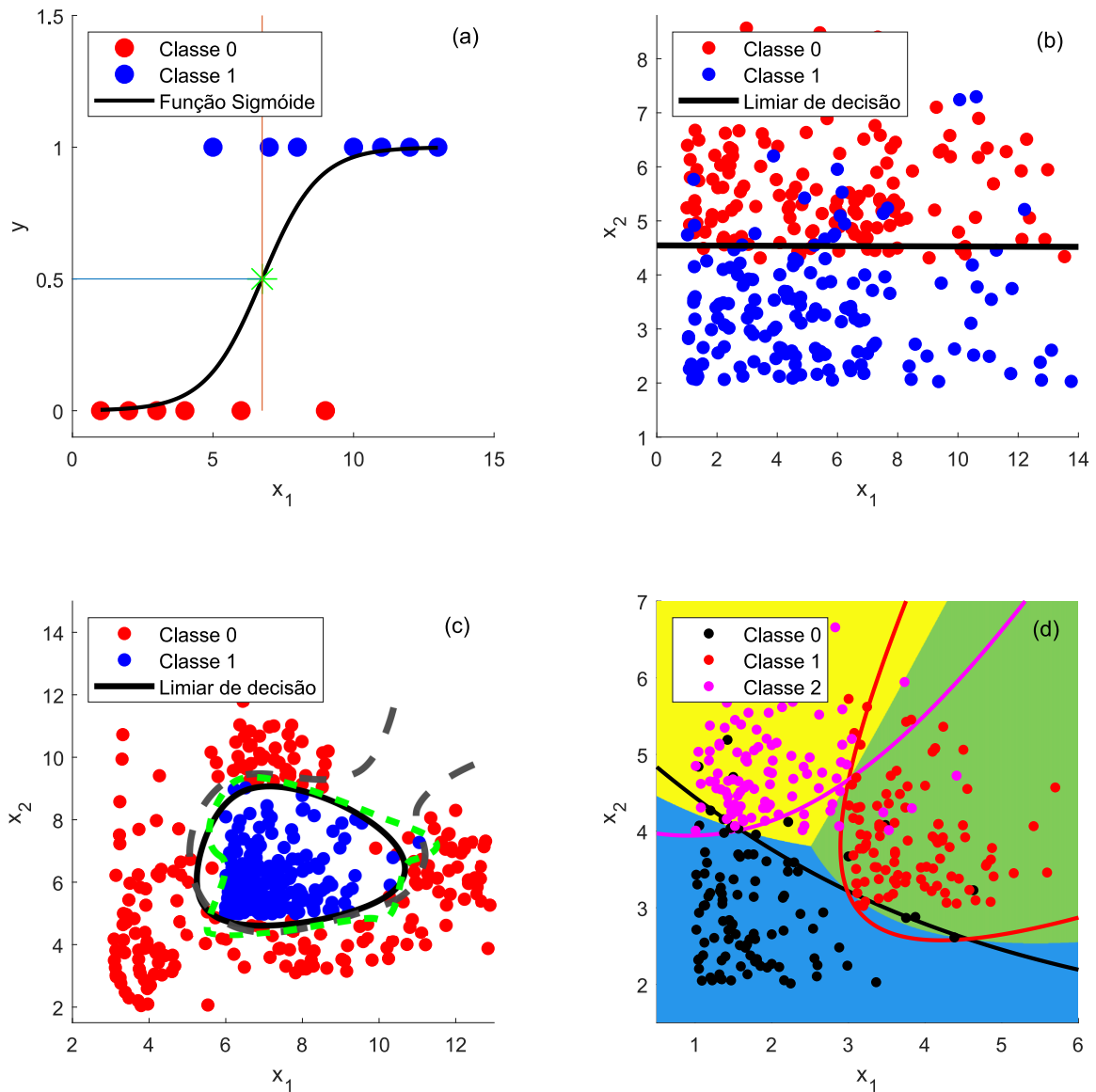


Figura 4 – Classificação por regressão logística para (a) um único Atributo  $x_1$ , (b) dois Atributos,  $x_1$  e  $x_2$ , (c) dois atributos porém com um limiar de decisão mais complexo, demonstrando *overfitting*, linha tracejada em verde, e *underfitting*, linha tracejada em cinza e (d) classificação em três classes .

A Figura 4(a) ilustra como a função hipótese é utilizada em regressão logística no caso de um modelo de uma única dimensão. Como mostrado a partir da equação 2.8, classifica-se positivo se a hipótese for maior ou igual a meio. O marcador em formato de asterisco, disposto em  $y = 0.5$  indica o limiar da classificação dos pontos, ou seja, pontos que possuem valores menores que  $y = 0.5$  são reconhecidos como negativos (0) e os que possuem um valor maior que este são classificados como positivos (1). A linha vertical define a fronteira de decisão, que é quando a função de hipótese está acima do limiar de  $h(x) = 0.5$ .

Na Figura 4(b) e (c), tem-se a representação de uma classificação ao se considerar dois atributos, isto é, duas dimensões, sendo exibido então em um plano. Na Figura 4(b) têm-se um arranjo simples na distribuição dos pontos. Uma função simples, a qual não utiliza o mapeamento polinomial de atributos, é capaz de definir bem o limiar de decisão entre as classes. No caso da Figura 4(c), se dá como necessário a utilização de polinômios de graus superiores para definir adequadamente tal fronteira.

A Figura 4(c) também ilustra o caso em que ocorre *underfitting*, representado pela curva tracejada em cinza. Tal caso demonstra um modelo que é simples demais para descrever a região pertencente a classe 1. A curva de cor preta da Figura 4(c) mostra um modelo ideal, em que a curva é bem adequada para a classificação. A Figura 4(c) também nos mostra o caso de *overfitting*, representado pela curva tracejada em verde. Neste caso pode-se observar que tal limiar de decisão é excessivamente complexo para descrever a região em que a classe pertence. Apesar do modelo representar bem os dados de treinamento, apresentará pouco desempenho a classificar dados futuros. Na Figura 4(c) tem-se que os diferentes cenários indicados são resultados de distintas configurações em relação ao termo de regularização e de mapeamento polinomial de atributos.

Na Figura 4(d) há três classes diferentes. Foi utilizado o método *One-versus-All*, em que cada uma das classes é classificada independentemente. A área da figura ilustrada em azul representa a região de decisão da classe 0, a área em verde é a região de decisão da classe 1 e a região em amarelo é a região de decisão da classe 2. É importante notar que as fronteiras entre as regiões de decisão não mais correspondem com os valores de  $h = 0.5$ , os limiares são obtidos considerando o maior valor da função hipótese entre as classes. Cada ponto que for adicionado na região em azul será classificado como classe 0, e o mesmo esquema é dado para as regiões de decisão em amarelo e em verde.

### 3 MÉTODOS

Os dados que representam os símbolos da constelação de um sistema digital coerente, os quais sofreram efeitos lineares e não lineares, foram obtidos por meio do software *VPI Transmission Maker*. Tais dados, em seguida, foram processados utilizando uma linguagem de alto nível, no caso, com a utilização do Matlab. O desempenho do algoritmo será avaliado por meio da taxa de erro de símbolo.

No *VPI Transmission Maker* se implementou um sistema digital coerente, pelo qual se transmitiu símbolos no domínio óptico. A simulação contabilizou tanto os efeitos lineares, como ruído e atenuação, como os efeitos não lineares, mais especificamente o *self-phase modulation*, causado pelo efeito Kerr. Posteriormente, os dados simulados foram processados no Matlab. Utilizou-se então um algoritmo de regressão logística, implementado por nós, a fim de compensar as não linearidades. A compensação de tais impedimentos tem o intuito de diminuir o valor da SER para potências de transmissão acima do limiar do efeito Kerr.

O sistema de comunicação óptica implementado no *VPI Transmission Maker* consiste em um enlace *single span* passivo de 100 km de comprimento. O sinal possui uma taxa de 56 Gbps e utiliza do formato de modulação 16-QAM. Próximo ao receptor, o sinal passa por um atenuador de 18 dB que representa a segmentação do enlace para 64 usuários. Este sistema modela um sistema PON de longo-alcance (*Long-Reach PON*, LR-PON).

#### 3.1 SISTEMA ÓPTICO COERENTE

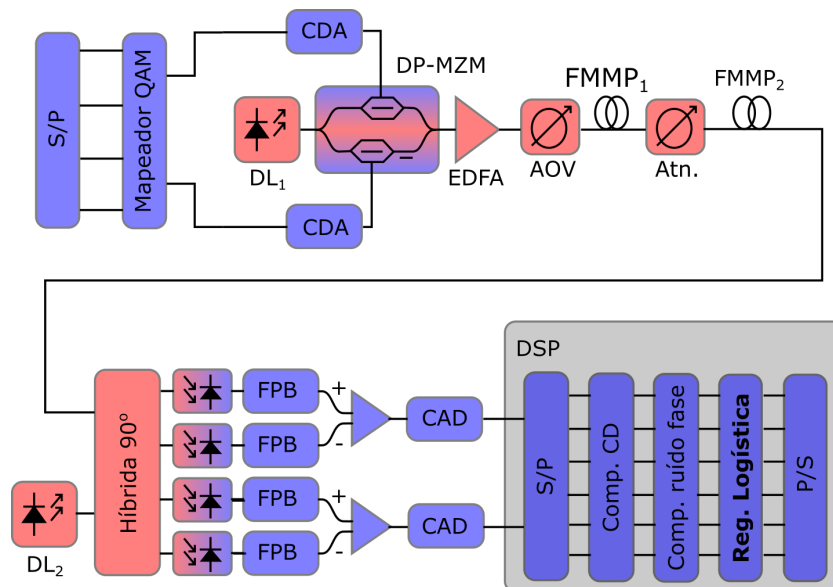


Figura 5 – Implementação do sistema LR-PON coerente. S/P: conversão de série para paralelo. CDA: conversor digital para analógico. DL: diodo laser. DP-MZM: moduladores *Mach-Zehnder* duplos em paralelo (*dual parallel- Mach-Zehnder modulator*). EDFA: amplificador de fibra dopada de érbio (*erbium-doped fiber amplifier*). AOV: atenuador óptico variável. FMMP: fibra óptica monomodo padrão. Atn.: atenuador. FPB: filtro passa-baixo. CAD: conversor analógico para digital. DSP: processamento digital de sinais.

A Figura 5 representa o diagrama de blocos para a simulação de um sistema LR-PON implementado no *software VPI Transmission Maker*. No transmissor, o conversor série-paralelo é responsável por reordenar os bits de forma a possibilitar o mapeamento para 16-QAM, que se dá por meio do mapeador QAM. Os dois conversores digitais-analógicos são responsáveis por gerar os sinais modulantes, um deles é reservado para a componente em fase e o outro para a componente em quadratura. A modulação do sinal é feita para o domínio óptico a partir da geração da onda de luz por um diodo laser. Tal diodo é conectado a uma estrutura dupla de moduladores MZM, necessita-se da operação de ambos para reproduzir a componente em fase e a em quadratura.

Os últimos blocos pertencentes ao transmissor, responsáveis por regular a potência óptica do sinal transmitido, são um amplificador por fibra dopada com Érbio (*Erbium-Doped Fiber Amplifier*, EDFA) e um atenuador óptico variável. É válido ressaltar que o EDFA e o atenuador óptico variável foram necessários na simulação no *software VPI*, apesar de que não são componentes convencionais em um sistema óptico coerente. O canal de comunicação se dá por meio de uma fibra óptica monomodo padrão (*Standard single-mode fiber*, SSMF) seguida de um atenuador de 18 dB, que corresponde a dividir o enlace óptico entre 64 usuários.

Por intermédio do segundo enlace SSMF, após a divisão da potência entre os usuários, se dá a detecção no receptor. Como dito na Seção 1.2, a detecção se dá de forma intradina, em que a onda do sinal faz o batimento com uma onda contínua gerada por um segundo laser, do qual dispõe frequências similares ao do sinal recebido. Por meio da rede híbrida, se permite a combinação do sinal recebido com o laser local com deslocamento de fase de 0, 90, 180 e 270 graus, possibilitando assim a recuperação das componentes em fase e em quadratura do sinal.

O sinal resultante de cada uma das quatro portas de saída da rede híbrida é fotodetectado, para então ser processado por meio de um filtro passa-baixas. Estes filtros dispostos após os fotodetectores representam suas respectivas bandas. Os sinais correspondentes a mistura do sinal com 0° e 180°, são subtraídos entre si para gerar a corrente descrita pela equação 1.10. Os sinais correspondentes às misturas com o laser deslocado em 90° e 270° são subtraídos entre si para gerar a corrente descrita pela equação 1.11.

Após a obtenção dos componentes em fase e em quadratura do sinal, ambos são convertidos para o domínio digital por meio de um conversor analógico-digital, para então se realizar o processamento digital de sinais. O primeiro bloco do processamento é um conversor série-paralelo a fim de reordenar os dados. O primeiro algoritmo a ser implementado tem a função de compensar a dispersão cromática do sinal sofrida pelo meio, seguido de um algoritmo capaz de compensar o ruído de fase causado pelo laser.

O último algoritmo a ser implementado no receptor, como se apresenta na Figura 5, é o de regressão logística, o qual representa o foco do nosso trabalho. Tal algoritmo é responsável por traçar as regiões de decisão na constelação que sofreu os efeitos da SPM. A classificação por regressão logística é seguida por um conversor paralelo-série, de modo a coordenar os dados por meio de um único enlace elétrico.

### 3.2 IMPLEMENTAÇÃO DO ALGORITMO DE REGRESSÃO LOGÍSTICA

Obtidos os dados em forma de constelação pelo sistema simulado no *software* VPI, a mitigação dos efeitos das não linearidades foi implementada no *software* Matlab. Simulou-se a transmissão de 81500 símbolos pelo sistema descrito pela Figura 5. Posteriormente foram gravados os dados que representam a componente real e a componente imaginária dos dados tanto no transmissor, quanto no receptor, após a propagação pelo canal não linear.

Primeiramente há duas entradas, representadas pela parte real e pela imaginária do símbolo recebido. Por meio da utilização de mapeamento polinomial de atributos, é possível mapear estas entradas em uma dimensão superior. Como é conhecido o símbolo que foi enviado, é possível definirmos as classes para os dados de treinamento. Implementa-se então a função hipótese, dada pela equação 2.6 e inicializa-se os parâmetros  $\theta$  com valores nulos. Introduce-se então a função de custo considerando a regularização, dada pela equação 2.16. Introduce-se também as atualizações dos parâmetros de  $\theta$ , dados pela equação 2.17, conhecidos como gradientes.

Como dito na Seção 2.3, com a utilização da equação 2.14 é possível encontrar o mínimo global da função de custo. Com uma função de custo convexa como é o caso da regressão logística, sua minimização é sempre garantida pelo *Gradient Descent*. O método chamado de *Gradient Descent* permite encontrar valores de  $\theta$  que satisfaçam a condição de minimização. Porém, por questões de otimização, o *Gradient Descent* não foi implementado diretamente. Utilizou-se de um pacote de otimização para o *MATLAB* conhecido como *Optimization Toolbox*, o qual contém uma função de interesse chamada *fminunc*.

Pela utilização da função *fminunc*, é possível encontrar o mínimo de uma função de múltiplas variáveis, assim como o valor das variáveis que resultam neste mínimo. Além da vantagem de ser capaz de resolver o problema mais rapidamente, não é necessário escolhermos um valor de  $\alpha$ , diminuindo assim a complexidade da configuração do problema.

A função *fminunc* recebe como entrada a função que se deseja minimizar, também se introduz o ponto de partida em que se deve inicializar o *fminunc*. O terceiro argumento é utilizado para especificar opções na função.

Para utilizar a função *fminunc*, primeiramente deve-se escolher algumas opções por meio da função *optimset*. Quando a opção chamada *GradObj* é configurada em *on* (ligada), *fminunc* retornará um segundo argumento. Tal argumento indica o gradiente, ou seja, a primeira derivada no ponto de mínimo. Outra opção que utilizamos foi a chamada de *MaxIter*, que estabelece a quantidade máxima de iterações realizadas para encontrar o mínimo de uma dada função.

Sendo assim, com a função *fminunc* retornamos quatro valores:

1. Valores de  $\theta$  em que a função de Custo é mínima.
2. O gradiente da função Custo em seu ponto de mínimo.
3. Um indicador de funcionamento denominado *exit flag*. Tal indicador terá o valor de -1 se o mínimo da função não foi encontrado com sucesso, o valor de 1 se foi encontrado com sucesso e o valor de 0 se o número máximo de iterações foi excedido antes de se encontrar o mínimo da função.



4. A quantidade de iterações realizadas para se encontrar o mínimo da função.

Introduz-se a função custo dada pela equação 2.16 como argumento da função *fminunc*, da qual se leva em consideração a regularização. Introduz-se também o gradiente da função custo, dado pela equação 2.17. A função *fminunc* realizará um processo conhecido como Otimização em Larga Escala (*Large-Scale Optimization*).

O algoritmo de *Large-Scale Optimization* é, principalmente, utilizado para problemas de *Big Data*, área da computação encarregada de processar e armazenar numerosas quantidades de dados. Tal algoritmo se mostra também muito eficiente em algoritmos como regressão logística e outros algoritmos de aprendizado de máquina. O algoritmo de *Large-Scale Optimization* utilizado para o problema é baseado nos métodos reflexivos de Newton, que realiza iterações através de trechos lineares em funções definidas em trechos (COLEMAN; LI, 1994). *Large-Scale Optimization* permite encontrar valores reais, sem restrições, que sejam capazes de minimizar uma função de múltiplas variáveis qualquer (COLEMAN; LI, 1994).

Para implementação do algoritmo de regressão logística em uma constelação 16-QAM, utilizou-se o método *One-versus-All*, citado na Seção 2.2. Dado que são 16 classes, cada uma representando um símbolo diferente, o algoritmo é executado em uma classe por vez. Considera-se a classe que está sendo classificada como a positiva, enquanto todas as outras são dadas como  $y = 0$ . Então os resultados obtidos por cada classificação pelo método *One-versus-All* são combinados, de forma a representar a classificação para as 16 classes.

## 4 RESULTADOS

### 4.1 AVALIAÇÃO DO DESEMPENHO DO ALGORITMO DE REGRESSÃO LOGÍSTICA PARA COMPENSAÇÃO DE NÃO LINEARIDADES

Como descrito no Capítulo 3, utilizando-se de um sistema óptico passivo *single-span* de 100 km e de um formato de modulação 16-QAM, analisou-se a SER obtida pelo sistema para níveis de potência desde 2 mW até 13 mW. A SER foi obtida tanto com a utilização de detecção por máxima verossimilhança, quanto por utilização de regressão logística.

A atenuação da fibra SSMF é de 0,2 dB/km. Pelo fato do enlace ter 100 km de comprimento, têm-se então 20 dB de atenuação causada pela fibra e 18 dB causado pelo atenuador, somando então em 38 dB de perdas. Com perdas de propagação tão elevadas, em combinação com uma taxa de transmissão de 56 Gbps, tem-se como resultado uma SNR baixa. Infelizmente, a potência lançada na fibra está limitada por não linearidades, desta forma, para o valor ótimo de LOP, o desempenho do sistema está limitado tanto por ruído como por distorção não linear.

A Figura 6 ilustra as constelações obtidas para os valores de potência desde (a) 2 mW até (l) 13 mW, com amplitudes normalizadas em relação à amplitude dos símbolos ideais de maior energia. Pode-se observar que, para potências baixas, de 2 mW à 4 mW, o sistema está limitado a ruído, devido à baixa potência do sinal recebido.

Pode-se ver na Figura 6(d) que, para um LOP de 5 mW, os símbolos da constelação apresentam uma boa definição. Observa-se também que, a partir de 7 mW, como mostra na Figura 6(f) até a Figura 6(l), a auto-modulação de fase resultante do efeito Kerr se torna cada vez mais significativa conforme a potência de transmissão é aumentada. Como discutido na Seção 1.3, isto é esperado, pois o efeito Kerr acentua-se com a intensidade da onda propagante no meio.

Nas constelações que sofrem o efeito Kerr da Figura 6 fica claro qual é a consequência da auto-modulação de fase. Como a fase é modulada, se vê uma rotação da constelação no plano complexo e os símbolos tendem a invadir a região dos símbolos vizinhos. Símbolos que estão mais distantes do centro, e portanto têm potências maiores, são mais afetados pela SPM. Isto se dá devido à dependência do efeito com a intensidade.

Na Figura 6 observa-se que os símbolos que estão mais perto do centro tendem a rotacionar no sentido horário. Isto ocorre devido ao algoritmo de correção de ruído de fase utilizado, o qual resulta em uma sobrecompensação nos símbolos mais próximos a origem do plano complexo. Para compensar-se o ruído de fase de símbolos mais distantes da origem, necessita-se de uma maior modificação do ângulo. Tal compensação causa este efeito de rotacionar excessivamente os símbolos de menor energia no sentido horário, porém tal efeito não é significativo o bastante para prejudicar o desempenho do sistema.

A Figura 7 tem como propósito comparar as regiões de decisão traçadas por máxima verossimilhança e por regressão logística para quando o sistema está operando nos casos extremos. Obtêm-se então os resultados para a condição em que o sistema está limitado por ruído aditivo e para quando o sistema está limitado pelo efeito não linear SPM.

A Figura 7(a) e a Figura 7(b) representam, respectivamente, as regiões de decisão geradas por

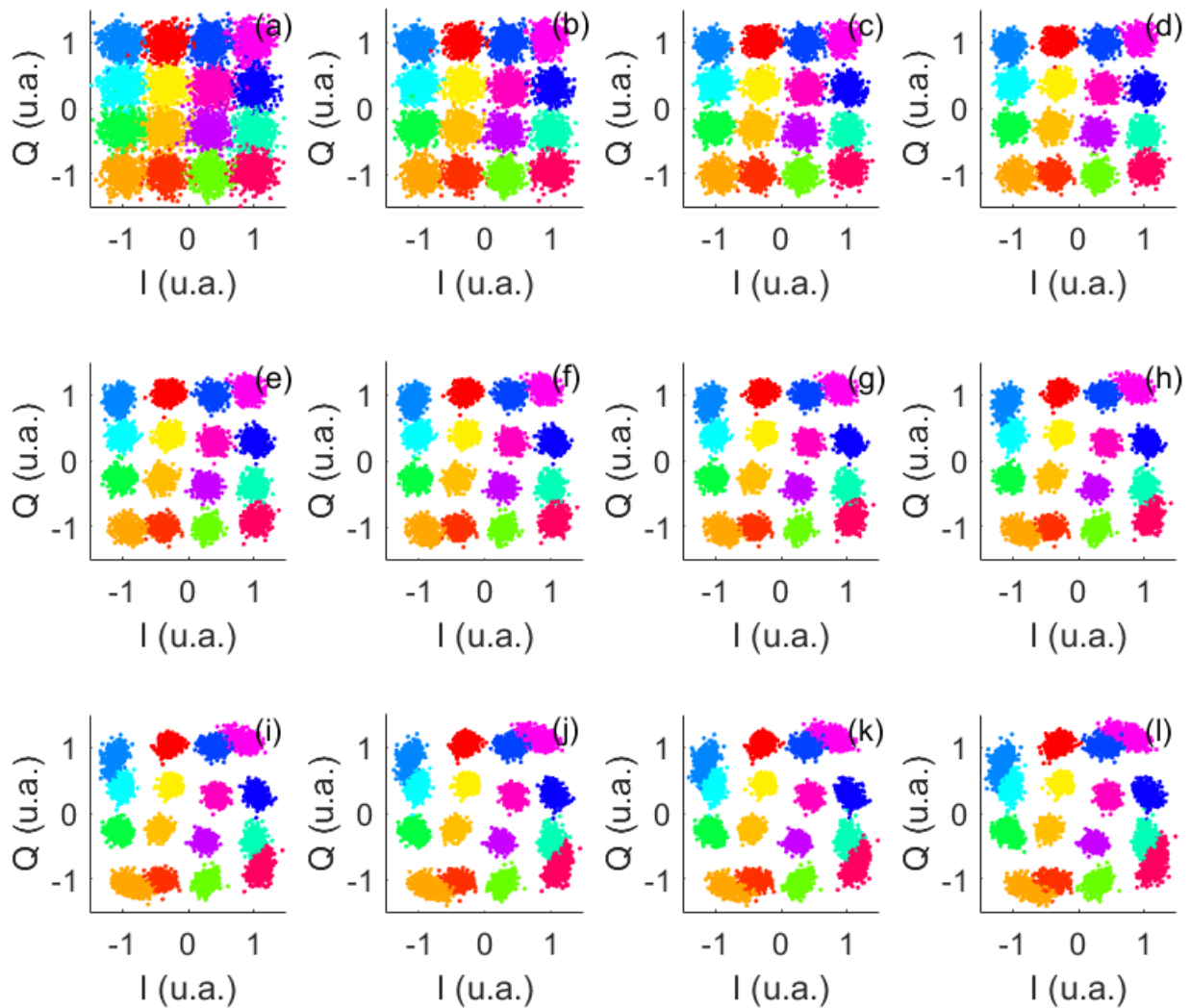


Figura 6 – Constelações de um sinal recebido utilizando modulação 16-QAM que sofreu efeito linear de ruído e efeito não linear de SPM com potências de transmissão variando desde (a) 2 mW até (l) 13 mW, com passo de 1 mW entre os itens da figura. Ainda não se aplicou o algoritmo de classificação, apenas se utilizou a classe de cada símbolo para representação por cores.

máxima verossimilhança e por regressão logística para um LOP de 2 mW. Percebe-se que não há muitas diferenças em relação as duas técnicas para quando o sistema está limitado por ruído aditivo. Isto ocorre pois os pontos apresentam uma distribuição aleatória, de modo que se espalham pelas regiões vizinhas e não há um padrão bem definido a se modelar. Mostra-se que as regiões traçadas pela regressão logística são semelhantes as traçadas pela máxima verossimilhança. A distribuição dos símbolos causada pelo ruído é aproximadamente gaussiana (AGRAWAL, 2014), sendo então os limiares traçados por máxima verossimilhança suficientes para descreve-los.

A Figura 7(c) e a Figura 7(d) ilustram, respectivamente, as regiões de decisão geradas por máxima verossimilhança e por regressão logística para um LOP de 13 mW. Pode-se observar que, com a

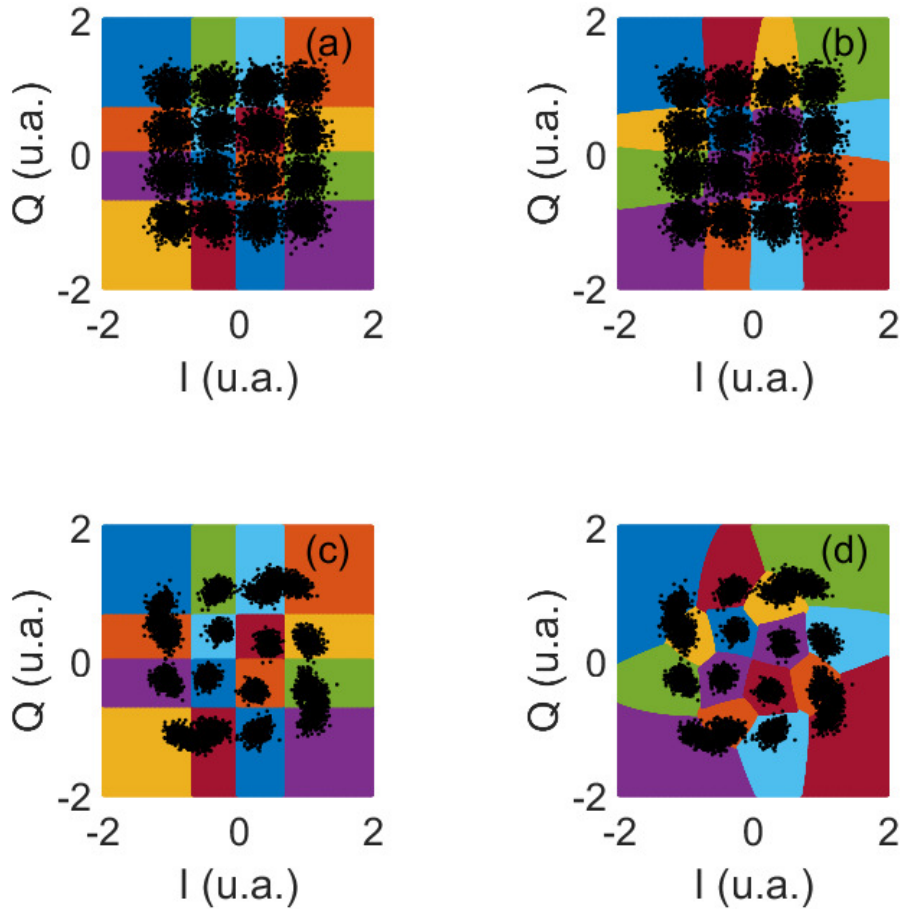


Figura 7 – Constelação de um sinal utilizando modulação 16-QAM com as regiões de decisão traçadas para um LOP de 2 mW (a) utilizando máxima verossimilhança e (b) utilizando regressão logística, e com um LOP de 13 mW (c) utilizando máxima verossimilhança e (d) utilizando regressão logística.

utilização de máxima verossimilhança, grande parte dos símbolos que possuem nível de energia mais alto se adentram na região do símbolo vizinho, resultando em erros significativos de decisão. Mesmo ao analisarmos um caso extremo como este, em que o efeito de SPM é dominante, o algoritmo de regressão logística ainda é capaz de reduzir satisfatoriamente as consequências negativas resultantes da SPM.

A partir da Figura 8 é possível demonstrar como são traçadas as regiões de decisão para máxima verossimilhança e para regressão logística em seu caso ótimo, ou seja, o LOP que gerou a menor SER entre eles. Como esperado, a regressão logística é capaz de se adaptar aos padrões resultantes do efeito Kerr, tendo as regiões de decisão traçadas de acordo com a rotação de fase na constelação utilizada para o treinamento do sistema. Os parâmetros de regressão logística utilizados para a obtenção dos resultados da Figura 7 e da Figura 8 foram um grau de polinômio de 2, obtido por meio da técnica de mapeamento polinomial de atributos discutida no Capítulo 2 e um valor de parâmetro de regularização  $\lambda = 0,02$ .

A Figura 9 mostra a SER, em escala logarítmica, em relação à valores de potência transmitida de 2 a 13 mW. Compara-se a utilização de máxima verossimilhança e a utilização de regressão logística

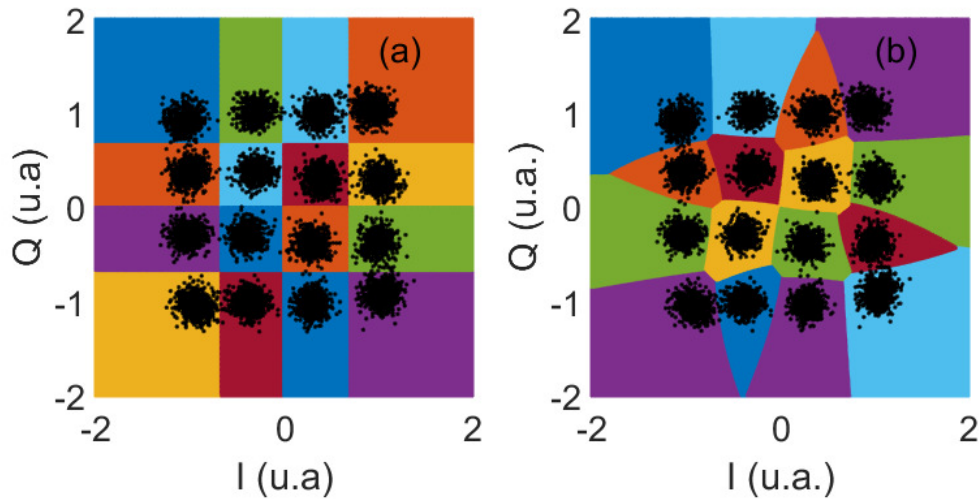


Figura 8 – Constelação de um sinal 16-QAM com as regiões de decisão traçadas para o caso ótimo utilizando (a) máxima verossimilhança, com 5 mW de LOP e (b) regressão logística, com 6 mW de LOP.

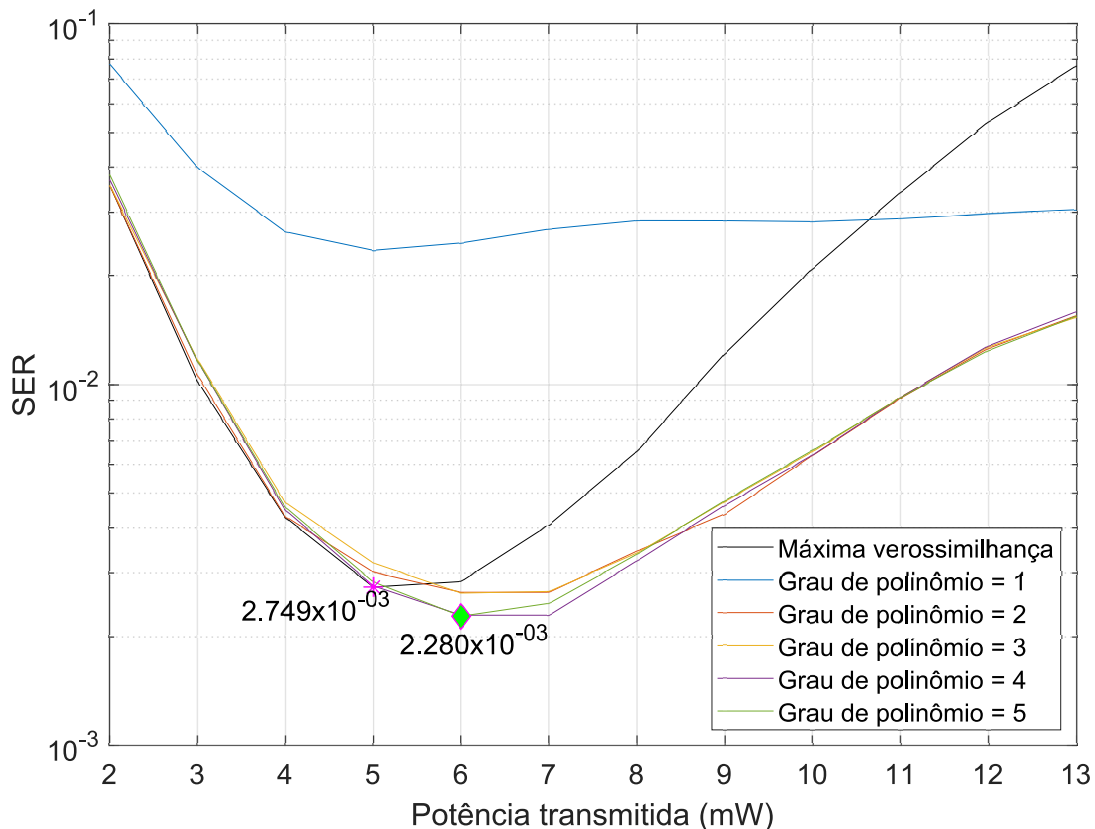


Figura 9 – Gráfico da SER em função da potência com a utilização de máxima verossimilhança e regressão logística utilizando regressão polinomial com grau de 1 a 5 com parâmetro de regularização  $\lambda = 0$ .

empregando mapeamento polinomial com grau de 1 a 5 e com  $\lambda = 0$ . Pode-se observar que a regressão logística apresenta um melhor desempenho na classificação que a máxima verossimilhança para um LOP com valores superiores a 5 mW, em que o efeito de SPM começa a se acentuar.

Na Figura 9 a configuração de regressão logística que mostrou o melhor desempenho foi com um grau polinomial igual a 4, indicado pela curva em roxo. É possível perceber que com a utilização de um grau de polinômio de 1, representado pela curva em azul, não foi possível obter resultados satisfatórios. Este fato indica que tal configuração não é capaz de modelar as regiões de decisão de forma adequada.

O marcador em formato de diamante na Figura 9 indica o menor valor de SER alcançado com a utilização de regressão logística que, como dito, ocorre para um valor de LOP de 6 mW. O marcador em formato de asterisco indica o valor mínimo de SER alcançado com a utilização de máxima verossimilhança, que ocorre para um LOP de 5 mW. Os resultados apresentados na Figura 9 nos mostra uma diferença de aproximadamente 5 erros a cada 10000 símbolos transmitidos entre os valores mínimos de SER obtidos com regressão logística e com máxima verossimilhança.

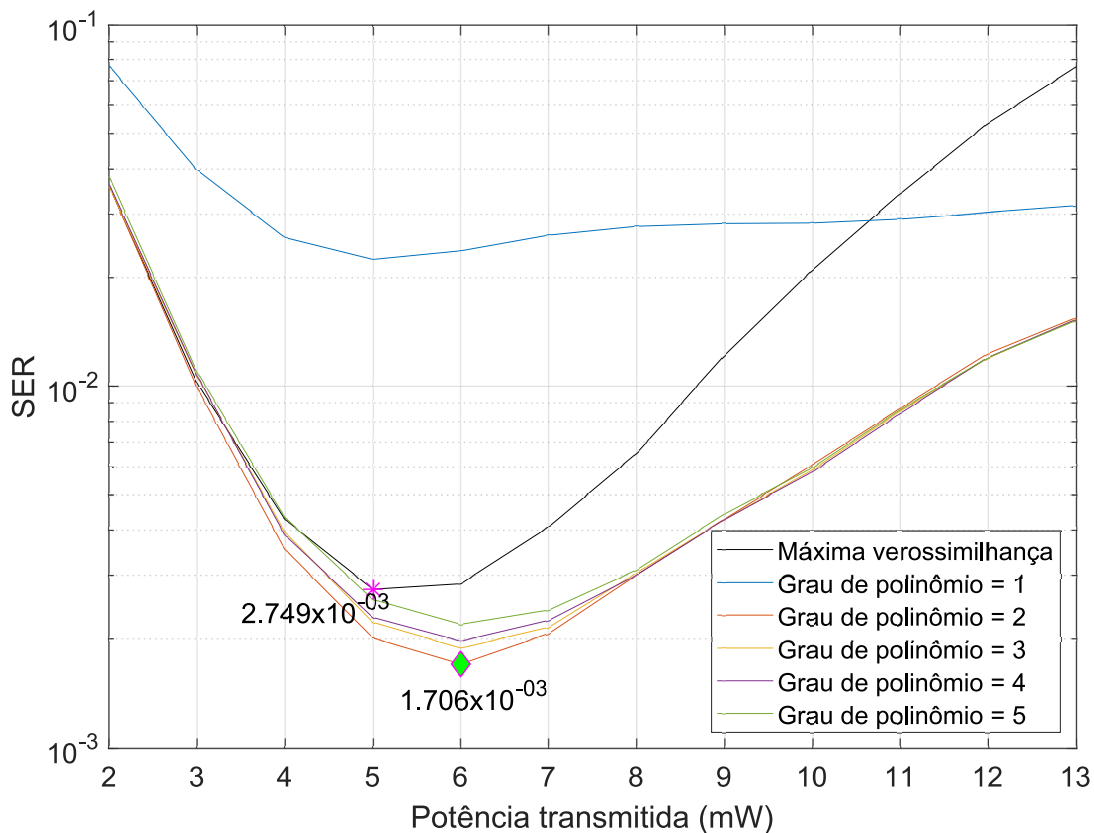


Figura 10 – Gráfico da SER em função da potência com a utilização de máxima verossimilhança e regressão logística utilizando regressão polinomial com grau de 1 a 5 com parâmetro de regularização  $\lambda = 0,02$ .

A Figura 10 indica o mesmo gráfico obtido na Figura 9 porém utilizando  $\lambda = 0,02$ , tal valor sendo a escolha para o parâmetro de regularização que demonstrou o melhor desempenho do algoritmo de regressão logística, representando então o valor ótimo para  $\lambda$ . Pode-se observar na Figura 10 que a SER ótima obtida por regressão logística apresenta uma taxa de aproximadamente 0,001 erros por símbolo a menos do que o valor de SER mínima obtida por máxima verossimilhança. concluindo então que, com a utilização de regressão logística, é possível obter uma redução de aproximadamente 38% da SER em comparação com a máxima verossimilhança.

## 4.2 REQUERIMENTOS DE SEQUÊNCIA DE TREINAMENTO

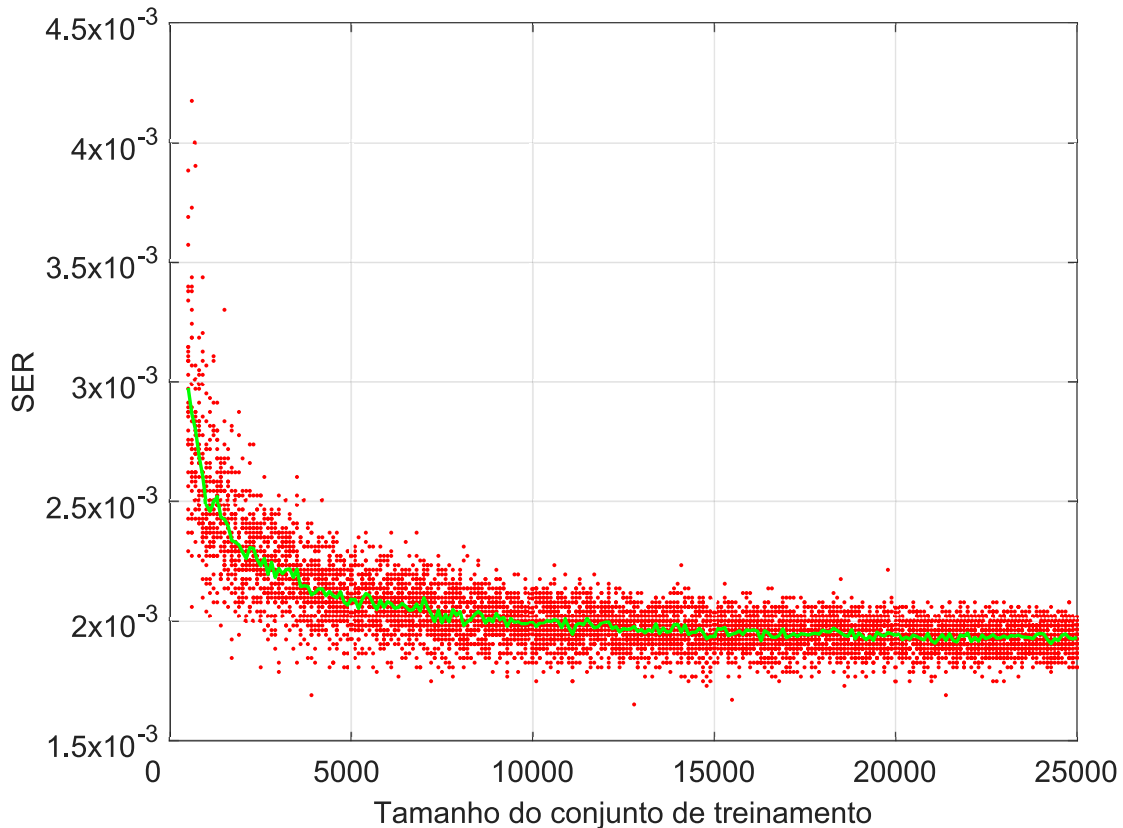


Figura 11 – Taxa de erro de símbolo em função do tamanho do conjunto de treinamento.

É válido definirmos a quantidade de amostras de treinamento em que o algoritmo mostre uma boa eficiência. A importância de utilizarmos a menor quantidade de dados de treinamento possível se dá pelo fato da necessidade de otimizarmos o tamanho do cabeçalho. Os dados utilizados para treinamento "desperdiçam" dados que poderiam ser utilizados para transmissão de informação. Quanto menos dados necessários para otimizarmos o treinamento, maior vai ser o aproveitamento para transmissão.

Uma outra vantagem de se utilizar um número elevado de pontos para o conjunto de validação no nosso trabalho é a de apresentar uma maior resolução ao mensurar o SER obtido. Têm-se então que os valores obtidos para o SER só podem tomar valores de múltiplos inteiros do SER mínimo. Dado um conjunto de validação com  $N_{\text{val}}$  pontos, o mínimo SER possível de se determinar é:

$$SER_{\text{min}} = 1/N_{\text{val}}. \quad (4.1)$$

A Figura 11 ilustra a SER em função do tamanho da amostra de dados utilizada como conjunto de treinamento. Para a análise utilizou-se um tamanho de amostra de treinamento entre 500 e 25000 exemplos, com um passo de 100 entre estes. Para cada tamanho de conjunto de treinamento, foi obtido o valor da SER para 30 amostras diferentes, sendo os dados aleatoriamente selecionados do conjunto total. O conjunto utilizado para o teste foi mantido fixo, com um tamanho de 56500 exemplos. Foi-se utilizado um  $\lambda = 0.02$  e um grau de polinômio de 2.

Na Figura 11 os pontos em vermelho indicam o valor da SER obtida para cada amostra, enquanto a curva em verde indica a média da SER obtida pelas amostras de um determinado tamanho. Se pode observar que, os resultados obtidos para a SER a partir de um conjunto de treinamento com aproximadamente 10000 exemplos atingem uma certa estabilidade, não apresentando muitas oscilações no valor da SER. Pode-se concluir então que 10000 exemplos para o treinamento é um tamanho satisfatório para modelar o problema.

### 4.3 AVALIAÇÃO DE COMPLEXIDADE

É importante definirmos uma métrica em relação a complexidade de computação do algoritmo, pois com isto podemos estimar o processamento gasto para a execução do mesmo. Têm-se que a complexidade do algoritmo de regressão logística utilizado no trabalho é totalmente dependente do grau de polinômio utilizado na mapeamento polinomial de atributos descrita pela equação 2.15.

A componente da complexidade relativa a mapeamento polinomial de atributos é estimada contabilizando a quantidade de Atributos por exemplo de treinamento gerados pela combinação entre os termos e também pelos produtos entre eles. Como, por exemplo, se considerarmos  $k = 2$  na equação 2.15, iremos ter dois termos de primeiro grau e três termos de segundo grau, contabilizando então um total de oito operações, provenientes de cinco somas e de três produtos. É desconsiderado aqui o atributo de *bias* pois seu valor é sempre constante e igual a 1.

Como forma de exemplificar, tem-se que sem a utilização do mapeamento polinomial de atributos há dois atributos de primeiro grau. Utilizando um grau  $k = 2$ , há então 8 operações, considerando as resultantes dos termos em primeiro e em segundo grau. O mesmo processo se dá ao utilizarmos um grau  $k = 3$ , em que obteríamos 20 operações. Pode-se então provar que a complexidade dada por um grau  $k$  é igual a:

$$C_1 = C_1(k) = \frac{1}{3}k^3 + k^2 + \frac{2}{3}k, \forall k \geq 1. \quad (4.2)$$

Deve-se considerar também a quantidade de elementos no vetor de parâmetros  $\theta$  necessários para modelar o sistema, que equivale a quantidade de elementos do vetor  $x^{(i)}$ . Para um grau de mapeamento polinomial de atributos  $k$ , multiplica-se cada um dos atributos estendidos pelos pesos dados pelos parâmetros  $\theta$ . Pode-se então provar que a quantidade de termos de  $\theta$  necessários para modelar o problema é dado por:

$$C_2 = C_2(k) = 1,5k + 0,5k^2 + 1, \forall k \geq 1, \quad (4.3)$$

em que a unidade somando é resultado do parâmetro  $\theta$  necessário para multiplicar a unidade de *bias*. Têm-se então que, ao utilizar um grau  $k = 1$ , necessitaríamos de apenas três parâmetros  $\theta$ , e ao utilizar um grau  $k = 2$ , necessitaríamos de seis parâmetros  $\theta$ . É necessário também contabilizar a quantidade de somas necessárias para gerar a entrada da função de ativação, que é dada por:

$$C_3 = C_3(k) = 1,5k + 0,5k^2, \forall k \geq 1. \quad (4.4)$$



A equação 4.4 descreve a quantidade de somas necessárias para se calcular a função de hipótese dado um grau  $k$ . Como estamos utilizando o método *One-versus-All*, tem-se que levar em consideração a quantidade de classes pertencentes ao problema, dado então que a complexidade  $C$  pode ser definida como:

$$C = C(k) = M[C_1 + C_2 + C_3] = M \left[ \frac{1}{3}k^3 + 2k^2 + \frac{11}{3}k + 1 \right], \forall k \geq 1. \quad (4.5)$$

Em que  $M$  representa a quantidade de classes utilizadas no problema de classificação. No trabalho estamos trabalhando com um problema de 16 símbolos, ou seja, há 16 classes, nos permitindo então re-escrever a equação 4.5 como:

$$C = 16 \left[ \frac{1}{3}k^3 + 2k^2 + \frac{11}{3}k + 1 \right], \forall k \geq 1. \quad (4.6)$$

Como forma de exemplificar, leva-se em conta os casos particulares de  $k = 3$ ,  $k = 4$  e  $k = 5$ , obtem-se então, respectivamente, um valor de complexidade computacional dado por  $C = 624$ ,  $C = 1104$  e  $C = 1776$ . A complexidade é dada em cálculos por símbolo classificado. Tais resultados mostram que, devido a função que descreve a complexidade computacional ser dada por um polinômio de terceira ordem, um único incremento no valor de  $k$  resulta em uma complexidade consideravelmente mais elevada.

A Figura 10 nos permite visualizar o fato de que, com a escolha adequada de um parâmetro de regularização, ao invés de tomá-lo como nulo como no caso da Figura 9, é possível atingir resultados mais satisfatórios. O grau 2 de polinômio foi o que apresentou a menor SER entre os utilizados, tal fato apresenta também a vantagem de ter relativamente pouca complexidade computacional. Como indicado na equação 4.6, a complexidade computacional para um grau 2 de polinômio é então de 304 operações por símbolo classificado. Por vias de comparação, ao utilizarmos um grau de polinômio de  $k = 5$ , tem-se uma complexidade de 1776 operações, ou seja, o algoritmo iria requerer de quase 5 vezes cálculos por símbolo classificado para sua execução.

## 5 CONCLUSÕES

No trabalho foram desenvolvidos métodos que nos permitiram mitigar as consequências introduzidas pelo efeito não linear de Kerr, sendo o foco do estudo o efeito de auto-modulação de fase. Uma transmissão feita a partir de um sistema óptico coerente digital *single-span* utilizando o formato de modulação 16-QAM foi simulado pelo software *VPI Transmission Maker*. A compensação dos efeitos não lineares foi feita através da implementação do algoritmo de regressão logística pelo *software* Matlab.

Introduziu-se os conceitos dos algoritmos baseados em regressão logística e analisou-se as configurações que geram o menor SER possível. O resultado obtido por tal configuração se mostrou significativamente melhor do que o SER obtido utilizando máxima verossimilhança. Não se encontraram estudos que fazem utilização do algoritmo de regressão logística para compensação de efeitos não lineares, como é o caso do efeito Kerr. Entretanto tal algoritmo se mostrou inovador, apresentando um desempenho satisfatório e uma baixa complexidade.

Ao utilizar a regressão logística, foi possível se alcançar uma redução de aproximadamente 38% da SER em relação a obtida pelo uso da detecção por máxima verossimilhança. Mostrou-se que, ao se utilizar da regressão logística, é possível transmitirmos uma potência de quase 8 mW mantendo a SER equivalente a ótima obtida ao utilizar um LOP de 5 mW com a técnica de Máxima verossimilhança. Um ganho como este na potência indica um maior alcance do enlace óptico.

Ajustou-se corretamente os parâmetros do algoritmo, como o grau de polinômio devido ao mapeamento polinomial de atributos e o termo de regularização. Sem a utilização de regularização, não se mostrou um ganho significativo ao usar regressão logística em relação à máxima verossimilhança. Ao encontrar um valor adequado para o termo de regularização, foi possível obter resultados satisfatórios e uma complexidade computacional relativamente baixa.

O trabalho trata de um sistema operando apenas por meio de um canal e considerando apenas uma polarização da luz. Trabalhos futuros deverão incluir a utilização de modulação por divisão de polarização e sistemas operando por múltiplos canais pela técnica de modulação WDM.

## REFERÊNCIAS

- AGRAWAL, G. **Sistemas de comunicação por fibra óptica**. [S.l.]: Elsevier Brasil, 2014. v. 4.
- ALPAYDIN, E. **Introduction to Machine Learning**. 2nd. ed. [S.l.]: The MIT Press, 2010. ISBN 026201243X, 9780262012430.
- CISCO. **Cisco Visual Networking Index: Forecast and Trends, 2017–2022 White Paper**. 2017. Disponível em: <<https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html>>.
- COLEMAN, T. F.; LI, Y. On the convergence of interior-reflective newton methods for nonlinear minimization subject to bounds. **Math. Program.**, Springer-Verlag New York, Inc., Secaucus, NJ, USA, v. 67, n. 1-3, p. 189–224, out. 1994. ISSN 0025-5610. Disponível em: <<https://doi.org/10.1007/BF01582221>>.
- FUKUMOTO, Y. et al. Experimental demonstration of spm compensation based on digital signal processing using a complex-valued neural network for 40-gbit/s optical 16qam signals. In: **2018 23rd Opto-Electronics and Communications Conference (OECC)**. [S.l.: s.n.], 2018. p. 1–2.
- GARMIRE, E. Perspectives on stimulated Brillouin scattering. **New Journal of Physics**, v. 19, n. 1, p. 011003, Jan 2017.
- HOSMER, D. W.; LEMESHOW, S. **Applied logistic regression (Wiley Series in probability and statistics)**. 2. ed. Wiley-Interscience Publication, 2000. ISBN 0471356328. Disponível em: <<http://www.amazon.com/Applied-logistic-regression-probability-statistics/dp/0471356328%3FSubscriptionId%3D192BW6DQ43CK9FN0ZGG2%26tag%3Dws%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3D0471356328>>.
- KAMINOW, I. P.; LI, T.; WILLNER, A. E. **Optical Fiber Telecommunications: Systems and Networks**. 5th. ed. Orlando, FL, USA: Academic Press, Inc., 2008. ISBN 0123741726, 9780123741721.
- KEISER, G. **Optical fiber communications**. [S.l.]: McGraw-Hill Singapore, 2010.
- KIM, J.-H. Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. **Computational Statistics & Data Analysis**, v. 53, n. 11, p. 3735 – 3745, 2009. ISSN 0167-9473. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0167947309001601>>.
- LATHI, B. P.; DING, Z. **Sistemas de comunicações analógicos e digitais modernos**. LTC, Rio de Janeiro, 2012.
- NG, A. Cs229 lecture notes - supervised learning. 2012.
- SAMUEL, A. L. Some studies in machine learning using the game of checkers. **IBM Journal on Research and Development**, v. 3, p. 210–229, 1959.
- SHALEV-SHWARTZ, S.; BEN-DAVID, S. **Understanding machine learning : from theory to algorithms**. Cambridge University Press, 2014. ISBN 9781107057135 1107057132. Disponível em: <[http://www.worldcat.org/search?qt=worldcat\\_org\\_all&q=9781107057135](http://www.worldcat.org/search?qt=worldcat_org_all&q=9781107057135)>.
- SUZUKI, N. et al. 100 gb/s to 1 tb/s based coherent passive optical network technology. **J. Lightwave Technol.**, OSA, v. 36, n. 8, p. 1485–1491, 2018.

THARWAT, A. Independent component analysis: An introduction. **Applied Computing and Informatics**, 2018. ISSN 2210-8327. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2210832718301819>>.

TORRES, J. G. et al. Characterization and compensation of induced non-linear phase offset in multicarrier optical systems based on clustering techniques. In: **20th Symposium on Signal Processing, Images and Computer Vision (STSIVA)**. [S.l.: s.n.], 2015.

YADAV, S.; SHUKLA, S. Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In: **2016 IEEE 6th International Conference on Advanced Computing (IACC)**. [S.l.: s.n.], 2016. p. 78–83.