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

**LUIS HENRIQUE CHIANG**


**Predição de potência de saída em amplificadores a fibra dopada com érbio baseada em redes neurais artificiais**


São João da Boa Vista
2021

**Luis Henrique Chiang**

# Predição de potência de saída em amplificadores a fibra dopada com érbio baseada em redes neurais artificiais

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

Orientador: Profº Dr. Ivan Aritz Aldaya Garde

São João da Boa Vista
2021

C532p

Chiang, Luis
Predição de potência de saída em amplificadores a fibra dopada com érbio baseada em redes neurais artificiais / Luis Chiang. -- São João da Boa Vista, 2021
34 p. : il., tabs.

Trabalho de conclusão de curso (Bacharelado - Engenharia de Telecomunicações) - Universidade Estadual Paulista (Unesp), Câmpus Experimental de São João da Boa Vista, São João da Boa Vista
Orientador: Ivan Aritz Aldaya Garde

1. Amplificadores de potência. 2. Redes neurais (Computação). 3. Telecomunicações. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca do Câmpus Experimental de São João da Boa Vista. Dados fornecidos pelo autor(a).

**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**

# MODELAGEM DE AMPLIFICADORES DOPADOS A ÉRBIO UTILIZANDO MACHINE LEARNING

Aluno: Luís Henrique Chiang
Orientador: Prof. Dr. Ivan Aritz Aldaya Garde

Banca Examinadora:

- Ivan Aritz Aldaya Garde (Orientador)
- Luiz Henrique Bonani do Nascimento (Examinador)
- Marcelo Luís Francisco Abbade (Examinador)

A ata da defesa com as respectivas assinaturas dos membros encontra-se no prontuário do aluno (Expediente nº 016/2020)

## AGRADECIMENTOS

A jornada foi longa, e com ela foi obtido diversos tesouros.

Agradeço à Deus por me dar o tesouro da confiança e da perseverança.

Agradeço aos colegas por me darem o tesouro da amizade e do trabalho em equipe.

Agradeço à todos os professores por me darem o tesouro do conhecimento e e da curiosidade.

Agradeço aos familiares pelo tesouro do suporte e do amor.

Agradeço aos meus amigos mais próximos, pelo encorajamento, companheirismo e espírito de luta.

Agradeço às dificuldades, pelo tesouro da superação de obstáculos.

Agradeço por todo esse caminho que trilhei, me dando o tesouro de experiência e emoções.

Agradeço à todos que diretamente ou indiretamente fizeram parte da minha vida, pelo tesouro de aprendizado com todos e com tudo.

**RESUMO**

Os amplificadores a fibra dopada com érbio representam um dispositivo crítico em muitos sistemas de comunicações. Estes dispositivos apresentam uma elevada potência de saída, uma figura de ruído relativamente baixa e uma sensibilidade baixa a polarização. Porém, na presença de múltiplos canais multiplexados em comprimento de onda, a concorrência pelos íons excitados de érbio resulta numa modulação cruzada de ganho. O modelo tipicamente adotado para considerar este efeito é o modelo de Giles-Desurvire que consiste num sistema de equações diferenciais acopladas. Neste trabalho propomos um modelo de amplificador a fibra dopada com érbio baseado em redes neurais artificiais capaz de considerar o efeito de ganho cruzado entre diferentes canais sem precisar integrar um sistema de equações diferencias acopladas. Em particular utilizamos um perceptron multicamada *feedforward* operando em regressão com uma única camada oculta para modelar cada uma das saídas do amplificador. Cabe mencionar que o número de neurônios na camada oculta é otimizado utilizando validação cruzada estocástica. Os resultados numéricos revelam que o modelo proposto prediz de forma precisa a potência de saída em um sistema de quatro canais. Também mostramos que a complexidade das redes neurais são diferentes para cada canal, sendo maior nos canais centrais. Para o objeto estudado, o custo computacional do modelo ANN treinado deste trabalho é de 370 operações de ponto flutuante, o qual é previsivelmente menor que a complexidade de resolver numericamente o modelo de Giles-Desurvire.

**PALAVRAS CHAVE: AMPLIFICADORES DE POTÊNCIA; TELECOMUNICAÇÕES; REDES NEURAIS (COMPUTAÇÃO).**

**ABSTRACT**

Erbium-doped fiber amplifiers represent a critical device in many communications systems. These devices feature a high output power, a relatively low noise figure, and a low sensitivity to polarization. However, in the presence of multiple wavelength-multiplexed channels, competition for the excited ions of erbium results in cross-gain modulation. The model typically adopted to consider this effect is the Giles-Desurvire model, which consists of a system of coupled differential equations. In this work we propose an erbium-doped fiber amplifier model based on artificial neural networks capable of considering the cross-gain effect between different channels without needing to integrate a system of coupled differential equations. In particular, we use a multilayer *feedforward* perceptron operating in regression mode with a single hidden layer to model each of the amplifier outputs. It is worth mentioning that the number of neurons in the hidden layer is optimized using stochastic cross validation. The numerical results reveal that the proposed model accurately predicts the output power in a four-channel system. We also show that the complexity of neural networks is different for each channel, being greater in the central channels. For the studied object, the computational cost of the ANN model trained in this work is 370 floating point operations, which is predictably less than the complexity of numerically solving the Giles-Desurvire model.

**KEYWORDS: POWER AMPLIFIERS, TELECOMMUNICATION, ARTIFICIAL NEURAL NETWORKS.**

## LISTA DE ILUSTRAÇÕES

## LISTA DE ABREVIATURAS E SIGLAS

AI - *Artificial intelligence*

ANN - *Artificial neural network*

EDFA - *Erbium doped fiber amplifiers*

MLP - *Multilayer perceptron*

OOK - *On-off keying*

SOA - *Semiconductor optical amplifier*

SNR - *Signal-to-noise ratio*

WDM - *Wavelength-division Multiplex*

# LISTA DE SÍMBOLOS

$P_{out}^{(k)}$ - potência de saída do k-ésimo canal

$P_{in}^{(k)}$ - potências de entrada do k-ésimo canal

$G$ - ganho do amplificador

$G_0\,(f_k)$ - ganho linear na frequência $f_k$

$P_{sat}(f_k)$ - potência de saturação na frequência $f_k$

$P_s^{(k)}(z)$ - potência do k-ésimo sinal centrado em $f_k$

$P_p(z)$ - potência do bombeio

$\Gamma_s$ - coeficiente de confinamento dos sinais

$\Gamma_p$ - coeficiente de confinamento do bombeio

$\sigma_s^e$ - secção de emissão do sinal

$\sigma_s^a$ - secção de absorção do sinal

$\sigma_p^e$ - secção de emissão da potência

$\sigma_p^a$ - secção de absorção da potência

$N_t$ - densidade populacional dos íons de érbio total

$N_2$ - densidade populacional dos íons de érbio excitados

$\alpha_s$ - coeficiente de atenuação dos sinais

$\alpha_p$ - coeficiente de atenuação do bombeio

$T_1$ - o tempo de emissão espontânea do estado excitado

$a_d$ - a área da secção transversal da região dopada da fibra

$h$ - a constante de Planck

$\nu_s$ - a frequência central da banda dos sinais

$\nu_p$ a frequência do feixe de bombeio

$W$ - vetor de pesos

$X$ - vetor de dados de entrada

$J$ - função de perda empírica

$\alpha$ - taxa de aprendizagem

**SUMÁRIO**

# 1 INTRODUÇÃO

## 1.1 EVOLUÇÃO DOS SISTEMAS DE COMUNICAÇÕES ÓPTICAS

A pesquisa em sistemas de comunicações baseados em fibra óptica começou na década de 1970, após o desenvolvimento dos *lasers* de semicondutor e das fibras de baixa perda (Agrawal 2016). Estes sistemas de comunicações ópticas podem ser divididos em cinco gerações, sendo cada uma caraterizada por um aprimoramento fundamental que resulta num incremento do produto capacidade-comprimento de enlace (Agrawal 2012).

A primeira geração de sistemas de comunicações ópticas operava na faixa de 0.8 $\mu$m e usava *lasers* de semicondutor construídos em GaAs e fibras multimodo de índice degrau, alcançando taxas de 45 Mb/s e comprimentos de enlaces entre repetidores opto-eletrônicos de 10 km. Este espaçamento entre repetidores, mesmo sendo muito baixo comparando com sistemas de comunicações modernos, representava uma melhora significativa com relação aos sistemas coaxiais que para a mesma taxa somente atingia 1 km. Evidentemente, maior espaçamento resultava num número menor de repetidores, reduzindo os custos de instalação e de manutenção da rede. Porém, o produto comprimento por capacidade dos sistemas da primeira geração estava limitado pela alta perda de transmissão da fibra (Agrawal 2016).

Afim de incrementar o produto capacidade-comprimento de enlace, a segunda geração de sistemas de comunicações por fibra óptica trabalhava na faixa de 1.3 $\mu$m, utilizando *lasers* de InP, em que a perda da fibra era menor que 1 dB/km. Embora para fibras multi-modo no início da geração tenha a taxa de transmissão limitada a 100Mb/s, após o desenvolvimento de fibras mono-modo, os sistemas de segunda geração passaram a ter uma taxa de transmissão de até 1.7 Gb/s com espaçamento entre repetidores de até 50 km (Agrawal 2016).

Visando reduzir ainda mais a perda de transmissão da fibra, desenvolveram-se *lasers* de alações quaternárias de InGaAsP, permitindo que os sistema de terceira geração operassem na janela de 1.55 $\mu m$, em que as fibras ópticas de sílica apresentam uma perda de transmissão mínima. Estes sistemas atingiram taxas de transmissão de 2.5 Gb/s e comprimentos de enlace entre 60 km e 70 km, sendo disponíveis comercialmente em 1990. Porém, mesmo operando na janela de miníma atenuação, os sistemas de terceira geração não conseguiram acompanhar o incremento produto comprimento por largura de banda requerido (Agrawal 2016).

Neste contexto, foram desenvolvidos os amplificadores ópticos a fibra dopada com érbio (EDFAs, do inglês *erbium-doped fiber amplifiers*) que permitem a amplificação simultânea de múltiplos canais na janela de 1.55 $\mu m$ multiplexados em comprimentos de onda (WDM, do inglês *wavelength division multiplexing*). Estes amplificadores deram inicio à quarta geração de comunicações ópticas caraterizada por sistemas WDM com transmissão multi-$span$ (Mears 1999).

Cabe mencionar que as quatro primeiras gerações de comunicações ópticas utilizavam principalmente o formato de modulação de chaveamento ligado-desligado (OOK, do inglês *on-off keying*), o qual apresenta uma eficiência espectral relativamente baixa. Para aumentar esta eficiência espectral, os sistemas de quinta geração fazem uso de receptores coerentes digitais. Desta forma, os receptores dos

sistemas de quinta geração tendem a apresentar uma melhor sensibilidade que os sistemas anteriores baseados em detecção direta. Porém, mesmo apresentando uma melhora significativa na sensibilidade do receptor, os sistemas de quinta geração continuam requerendo EDFAs, ao menos nos sistemas de longa distancia e metropolitanos (Agrawal 2016).

## 1.2 AMPLIFICADORES ÓPTICOS

Como mencionado anteriormente, os EDFAs desempenham um papel crítico numa grande parte dos sistemas de comunicações de alta capacidade modernos. Porém, os EDFAs não são a única tecnologia de amplificação disponível. De fato, também existem amplificadores de semicondutor (SOAs, do inglês *semiconductor optical amplifiers*), amplificadores paramétricos e amplificadores Raman. A pergunta que naturalmente emerge é o que faz com que os EDFAs sejam a tecnologia de amplificação chave nos sistemas de comunicações?

Os amplificadores ópticos de semicondutor (SOA, do inglês *semiconductor optical amplifiers*) foram os primeiros amplificadores ópticos desenvolvidos, pois compartilham grande parte da tecnologia dos lasers (Dutta e Wang 2013). Porém, estes amplificadores apresentam uma figura de ruído alta, em torno a 9 dB, uma potência de saída máxima baixa e banda de operação relativamente estreita, além de uma alta distorção não linear e um ganho altamente dependente do estado de polarização do sinal de entrada. Essas limitações fazem com que os SOAs não sejam uma solução viável como amplificador em linha em sistemas de comunicações ópticas, principalmente em WDM.

Os amplificadores paramétricos exploram algum efeito paramétrico físico, principalmente o efeito Kerr, para conseguir converter fótons de um sinal de bombeio à frequência do sinal a amplificar (Baumgartner e Byer 1979). Estes amplificadores representam uma alternativa para incrementar a potência de sinais em comprimentos de onda nas quais os EDFAs não conseguem operar. Infelizmente, a implementação dos amplificadores paramétricos é complicada e tipicamente requer fibras altamente não lineares com perfis de dispersão muito precisos. Consequentemente, este tipo de amplificador também não é uma opção viável para sistemas de comunicações comerciais (Marhic 2008).

Os amplificadores Raman operam utilizando o espalhamento Raman estimulado das fibras. É um tipo de amplificação que apresenta uma figura de ruído idealmente nula e um ganho com pequena dependência do estado de polarização (Agrawal 2012). Além disso, os amplificadores Raman podem usar a própria fibra de propagação como meio de ganho. A única desvantagem dos amplificadores Raman é o limitado ganho, pois este fica saturado pela perda de propagação da fibra que reduz a potência do bombeio. Devido ao relativo baixo ganho, os amplificadores Raman geralmente são usados em combinação com os amplificadores EDFAs (Bromage 2004).

Finalmente, os amplificadores EDFAs apresentam um ganho elevado, uma figura de ruído relativamente baixa (de 4 a 6 dB), uma baixa dependência à polarização de entrada, uma banda de operação larga, uma potência de saturação de saída alta e uma complexidade relativamente baixa. Estas caraterísticas, junto com uma tecnologia suficientemente madura, fazem com que os amplificadores EDFAs sejam realmente uma peça fundamental em muitos sistemas de comunicações ópticas. A Tabela 1 apresenta uma comparativa das diferentes tecnologias de amplificação em termos de diferentes métricas (15).

| | Tipo | Ganho | Ruído | Faixa de operação | Saturação | Polarização | Vantagem | Desvantagem |
|---|---|---|---|---|---|---|---|---|
| SOA | discreto | baixo | 9 dB | limitado por material | baixo | sensível | integrável | baixa performance |
| Raman | distribuído | moderado | 0 dB (idealmente) | virtualmente ilimitado | moderado | insensível | fibra de propagação é utilizada | requer múltiplos lasers |
| Paramétrico | distribuído | moderado | 0 dB (idealmente) | virtualmente ilimitado | moderado | sensível | fibra de propagação é utilizada | requer fibras especiais / Brillouin |
| EDFA | discreto | alto | 4-6 dB | limitado por dopante | alta | insensível | maduro e fácil de entender | dependente do dopante |

Tabela 1 – Tabela comparativa das diferentes tecnologias de amplificação.

## 1.3 AMPLIFICADORES A FIBRA DOPADA COM ÉRBIO

Os amplificadores EDFA, como o próprio nome indica, utilizam uma fibra dopada com érbio (de fato, podem se utilizar outros elementos de terras raras como túlio e itérbio também). Alguns dos íons de terra rara presentes no substrato de sílica são excitados utilizando bombeio óptico, geralmente com um comprimento de onda de 1.490 $\mu$m ou 980 nm (Desurvire e Zervas 1995). Desta forma, se um fóton de sinal interage com um íon excitado, este decai por meio de um evento de emissão estimulada. Se o número de fótons e de íons excitados é suficientemente alto, a probabilidade de emissão estimulada supera a probabilidade de emissão espontânea e obtém-se ganho óptico. Desta forma, em um EDFA, o ganho experimentado por um sinal depende de dois fatores principais (Agrawal 2012; Desurvire e Zervas 1995):

- Por um lado, depende das secções transversais de captura e emissão dos íons de terra rara, os quais por sua vez dependem da estrutura de bandas dos íons na matriz de sílica. Estas secções transversais são proporcionais às probabilidade de absorção e emissão de fótons e, portanto, estão associadas à perda e ao ganho óptico. Devido a sua dependência das bandas do íons, é importante notar, que estas secções transversais variam significativamente com o comprimento de onda. De

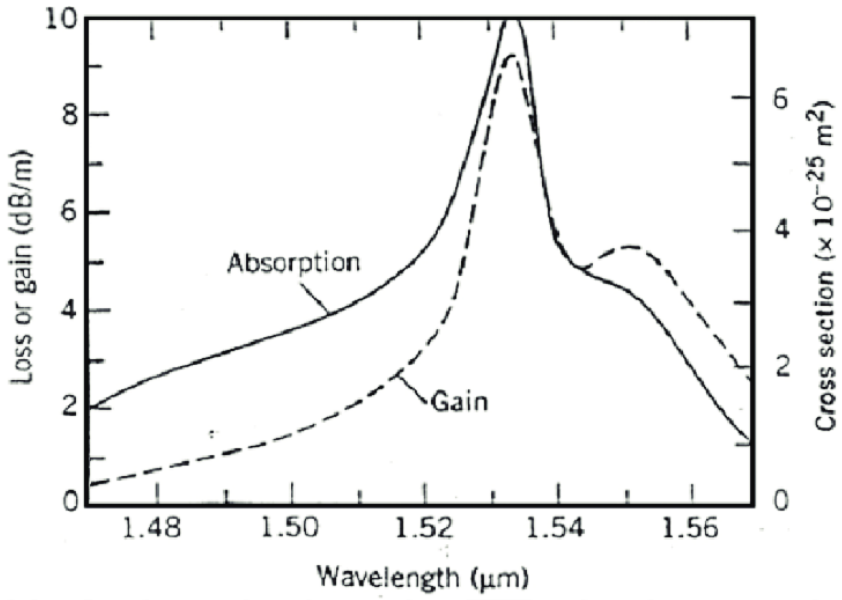


Figura 1 – Espectro de absorção e ganho de um EDFA (Agrawal 2012).

fato, são estas secções as que em grande parte limitam a banda de operação dos amplificadores. Além disso, também fazem com que o ganho não seja constante na banda de operação mas que tenha a forma característica mostrada na Figura 1. Desta forma, alguns canais WDM serão mais amplificado que outros, causando uma desigualdade nas potências (Agrawal 2012).

- Por outro lado deve-se considerar que o conjunto de íons excitados é compartilhado pelos diferentes sinais de entrada em sistemas WDM. Isto é, os diferentes sinais de entrada concorrem entre si para aproveitar esses íons como fonte de emissão estimulada (Menif et al. 2001).

Estas duas caraterísticas fazem com que o ganho experimentado por um determinado sinal dependa não somente de sua própria frequência e potência, mas também das potencias e frequências dos outros sinais. Desta forma, predizer o ganho de um determinado sinal não e trivial. Este cálculo é ainda mais complicado em sistemas multicanais com diferentes potências por canal, um cenário típico em redes metropolitanas e de fácil acesso nas quais o tráfego é tipicamente mais heterogêneo que em enlaces de longa distância (Zimmerman e Spiekman 2004; Wagner et al. 2016). Neste contexto tem-se proposto diferentes modelos de EDFAs.

## 1.4 MODELOS DE AMPLIFICADORES A FIBRA DOPADA COM ÉRBIO

Os modelos de EDFA podem se classificar em modelos dinâmicos e modelos estacionários. Os primeiros consideram a dinâmica dos íons, o que inclui tanto a geração como a decaimento dos íons excitados. Estes modelos são requeridos para modelar a operação em que os sinais de entrada variam em tempo mais lento que o tempo de vida médio dos íons excitados, i.e. $\approx$1 ms (Giles e Desurvire 1991). Por tanto, os modelos dinâmicos são utilizados em aplicações de pulsos curtos o cumpridos mas com um ciclo de trabalho baixo. Em sistemas de comunicações ópticas modernos, em que as taxas de transmissão excedem dezenas de Gbps, o período de símbolo é significativamente menor que o tempo caraterístico dos íons de érbio e podemos considerar as potências médias. Neste caso, os denominados modelos estacionários podem ser aplicados. Por sua vez, modelos estacionários com complexidade crescente podem ser identificados.

### 1.4.1 Modelo de ganho independente da frequência e da potência

O modelo mais básico desconsidera a dependência do ganho tanto da potência como da frequência do sinal a amplificar (Agrawal 2012). Desta forma, a potência de entrada e saída estão relacionadas por uma constante conforme à equação:

$$P_{out}^{(k)} = G \cdot P_{in}^{(k)}, \tag{1.1}$$

sendo $G$ o ganho e $P_{out}^{(k)}$ e $P_{in}^{(k)}$ as potências de saída e entrada do k-ésimo canal(na k-ésima frequência ), respetivamente. Mesmo sendo um modelo simples, pode ser utilizado em sistemas de um único canal com uma potência de saída menor que a potência de saturação.

### 1.4.2 Modelo de ganho dependente da frequência e da potência

O seguinte modelo considera a dependência do ganho com relação à frequência e potência do próprio sinal a ser amplificado[1] (Agrawal 2012). Desta forma, as potências de saída e de entrada estão relacionadas por:

$$P_{out}^{(k)} = G\left(f_k, P_{in}^{(k)}\right) \cdot P_{in}^{(k)} \text{ com } G\left(f_k, P_{in}^{(k)}\right) = \frac{G_0\left(f_k\right)}{1 + \frac{P_{in}^{(k)}}{P_{sat}(f_k)}} \tag{1.2}$$

em que $G_0(f_k)$ é o ganho linear na frequência $f_k$ e $P_{sat}(f_k)$ é a potência de saturação também na frequência $f_k$. Cabe destacar que temos modelos de complexidade intermediaria entre o modelo I e o modelo II. Se forçamos $G_0(f_k)$ e $P_{sat}(f_k)$ a $G_0$ e $P_{sat}$, obtemos um modelo que depende da potência do próprio sinal mas que desconsidera a dependência frequencial, o qual é útil em sistemas mono-canais com potências próximas à saturação. Alternativamente, considerando uma potência de saturação alta, o modelo se simplifica, considerando unicamente a dependência frequencial. Este modelo é amplamente utilizado em sistemas WDM multi-canais com potências relativamente baixas.

### 1.4.3 Modelo de Giles-Desurvire

Finalmente o último modelo considera não somente a frequência e a potência do próprio sinal mas também as frequências e potências dos outros canais (Giles e Desurvire 1991). Este modelo é comumente denominado de modelo Giles-Desurvire e pode ser expressado por meio do seguinte sistema de equações diferenciais que descreve a evolução espacial na direção de propagação $z$ das potências dos sinais, bombeio e a densidade de íons excitados:

$$\begin{aligned}
\frac{\partial P_s^{(k)}(z)}{\partial z} &= \Gamma_s\left[\sigma_s^e(f_k)N_2 - \sigma_s^a(f_k)(N_t - N_2)\right]P_s^{(k)}(z) - \alpha_s P_s^{(k)}(z) \\
\pm\frac{\partial P_p(z)}{\partial z} &= \Gamma_p\left[\sigma_p^e(f_k)N_2 - \sigma_p^a(f_k)(N_t - N_2)\right]P_p(z) - \alpha_p P_p(z) \\
N_2(z) &= -\frac{T_1}{a_d h\nu_s}\cdot\frac{\partial}{\partial z}\left[\sum_{k=1}^{N_{ch}} P_s^{(k)}\right] \pm \frac{T_1}{a_d h\nu_p}\frac{\partial P_p(z)}{\partial z},
\end{aligned} \tag{1.3}$$

sendo:

- $P_s^{(k)}(z)$ a potência do k-ésimo sinal centrado em $f_k$ e $P_p(z)$ a potência do bombeio;
- $\Gamma_s$ e $\Gamma_p$ os coeficientes de confinamento dos sinais e do bombeio respectivamente;
- $\sigma_s^e$, $\sigma_s^a$, $\sigma_p^e$ e $\sigma_p^a$ as secções de captura e emissão dos sinais e do bombeio;
- $N_t$ e $N_2$ as densidades populacionais dos íons de érbio total e excitados;
- $\alpha_s$ e $\alpha_p$ os coeficientes de atenuação dos sinais e do bombeio;
- $T_1$ o tempo de emissão espontânea do estado excitado;
- $a_d$ a área da secção transversal da região dopada da fibra;
- $h$ a constante de Planck e
- $\nu_s$ e $\nu_p$ a frequência central da banda dos sinais e do feixe de bombeio, respetivamente.

[1] É importante ressaltar que nos referimos a sinal a cada um dos canais WDM.

Como pode-se observar, este sistema acopla as potências dos sinais e de bombeio. Por outro lado, os sinais $+$ e $-$ permitem considerar a bombeio co-propagante e contra-propagante.

Desta forma, podemos concluir que temos modelos facilmente configuráveis mas que não modelam de forma precisa a interação entre os diferentes sinais. Por outro lado, o modelo de Giles-Desurvire é o mais preciso pois considera as frequências e potências de todos os sinais que interagem. Porém, o modelo requere da caraterização de múltiplos parâmetros que pode ser difícil de extrair num cenário real.

## 1.5 PROBLEMA DE PESQUISA

O modelo de Giles-Desurvire representa mais uma ferramenta de simulação e projeto que um método para modelar EDFAs reais. Ainda mais, considerando que a maioria dos EDFAs estão compostos por vários estágios de amplificação, o número de parâmetros a estimar é extremadamente alto. O modelo de Giles-Desurvire também apresenta desafios em termos de simulação, pois requere resolver um sistema de equações diferenciais. Desenvolver um modelo para um amplificador que considere de forma precisa a dependência frequencial e a interação entre os diferentes sinais de informação é necessário principalmente em redes metropolitanas com níveis de potência irregulares. Além disso, é desejável que possa modelar EDFAs realistas sem precisar uma caraterização detalhada dos parâmetros.

## 1.6 CONTRIBUIÇÃO DO TRABALHO

No presente trabalho, desenvolvemos um modelo de EDFA baseado em redes neurais artificiais (ANNs, do inglês *artificial neural networks*). Para comprovar e quantificar a precisão do modelo proposto utilizaremos resultados de simulação dum EDFA de duplo estágio obtidos mediante VPI Transmission Maker que implementa o modelo de Giles-Desurvire. Além de propor o modelo, otimizaremos o número de neurônios da camada oculta a fim de obter um modelo mais preciso possível.

## 1.7 ORGANIZAÇÃO DO DOCUMENTO

Neste trabalho foi implementado algoritmos de ANN para predição de potência de saída em amplificadores a fibra dopada com érbio. O documento é dividido em:

- O Capítulo 2 apresenta os princípios de redes neurais artificiais e do funcionamento dos perceptrons. Explica também diferentes classificações para diversas arquiteturas de rede neurais.
- O Capítulo 3 descreve o modolo do de EDFA utilizado em VPI Transmission Maker para obter os dados a serem processados.
- O Capítulo 4 apresenta os resultados utilizando ANN para predição das potências de saídas do amplificador, comparando os valores dos *outputs* da rede com os valores dos dados.
- Por fim, o Capítulo 5 destaca as principais conclusões obtidas e sugere possíveis trabalhos e estudos que podem ser realizados futuramente.

# 2 PRINCÍPIOS DE REDES NEURAIS ARTIFICIAIS

Uma rede neural é uma estrutura matemática computacional que é baseada em sistema de comunicação do cérebro humano (Anthony e Bartlett 2009; Wang 2003; Alpaydin 2009). O cérebro humano possui um conjunto extremamente complexo de células, os neurônios. Eles têm um papel essencial na determinação do funcionamento e comportamento do corpo humano e do raciocínio. Os neurônios são formados pelos dendritos, que são um conjunto de terminais de entrada, pelo corpo central, e pelos axônios que são longos terminais de saída. Os neurônios se comunicam através de sinapses. Sinapse é a região onde dois neurônios entram em contato e por meio da qual os impulsos nervosos são transmitidos entre eles. Os impulsos recebidos por um neurônio A, em um determinado momento, são processados, e atingindo um dado limiar de ação, o neurônio A dispara, produzindo uma substância neurotransmissora que flui do corpo celular para o axônio, que pode estar conectado a um dendrito de um outro neurônio B. O neurotransmissor pode diminuir ou aumentar a polaridade da membrana pós-sináptica, inibindo ou excitando a geração dos pulsos no neurônio B. Este processo depende de vários fatores, como a geometria da sinapse e o tipo de neurotransmissor. Mesmo se desde o ponto de vista biológico um neurônio é um sistema complexo, matematicamente pode ser descrito de forma relativamente simples.

## 2.1 NEURÔNIO ARTIFICIAL

O modelo mais amplamente utilizado para emular um neurônio de forma artificial é denominado perceptron. Mas o que um perceptron faz? Ele é uma unidade de processamento básico que realiza operações com as informações de entrada e retorna com uma informação da saída (Alpaydin 2009). Um perceptron é uma unidade computacional simples, porem quando juntamos vários perceptrons em uma arquitetura funcional, criamos uma rede que consegue processar informações de forma altamente flexível e complexa. O processamento num perceptron pode ser dividido em três etapas, tal e como mostrado na Figura 2.

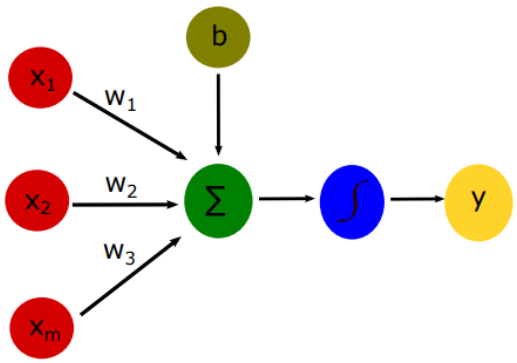


Figura 2 – Modelo perceptron de um neurônio.

Na primeira etapa, é realizada a multiplicação dos valores de entrada com seus respectivos pesos. Isso representa o quanto cada informação da entrada vai influenciar, ou pesar, na saída do perceptron.

Se chamamos as informações de entrada de $x_i$ e os pesos $w_i$ onde $i$ varia de $1, 2, ..m$, teremos uma somatória dada por $\sum_{i=1}^{m} x_i \cdot w_i$. Também podemos representar essa somatória como um produto escalar representado por $W \cdot X^t$, sendo $X = \{x_1, x_2, ...x_m\}$ o vetor de entradas e $W = \{w_1, w_2, ...w_3\}$ o vetor de pesos.

A segunda etapa é a adição do termo de bias *b* (as vezes denominado de polarização). O termo de bias é uma constante adicional que ajuda o modelo da rede neural a representar a sua função em padrões que não necessariamente passa pelo ponto de origem, como grande parte das funções reais. Juntamente com a primeira parte, podemos representar a soma como:

$$z = W \cdot X^t + b. \tag{2.1}$$

A terceira etapa é processar a variável $z$ utilizando uma função de ativação. Até a segunda etapa temos uma função linear, possibilitando o modelo de resolver funções linearmente separáveis. Porém no mundo real, raramente os dados são linearmente separáveis, fazendo com que tenha a necessidade de uma transformação não-linear. Há vários diferentes tipos de funções de ativação, cada um adequado para seus modelos e objetivos diferentes. Nesse trabalho será utilizada a função não-linear sigmoide $sig$[1], a qual está dada por:

$$sig(z) = \frac{1}{1 + \exp(-z)}. \tag{2.2}$$

Uma das vantagens desta função é que qualquer valor real de entrada é mapeado na faixa entre [0 e 1]. Além disso, esta função é amplamente utilizada por ter uma derivada com expressão fechada, o qual facilita o cálculo do gradiente. Por outro lado, a desvantagem de usar esta função, é que nos seus extremos, os valores de saída tendem a apresentar uma sensibilidade menor aos valore de $z$ e por conseguinte, também aos elementos de $X$. Por fim, podemos representar a saída do perceptron da seguinte maneira:

$$y = sig(z) = sig(W \cdot X^t + b). \tag{2.3}$$

## 2.2 REDES NEURAIS ARTIFICIAIS

Uma vez entendido o funcionamento dum único perceptron, estudaremos diferentes arquiteturas em que estes podem ser interligados. Consideremos a rede neural artificial mostrada na Figura 3.

[1] Estritamente Esta função também é denominada de função logística em alguns textos.

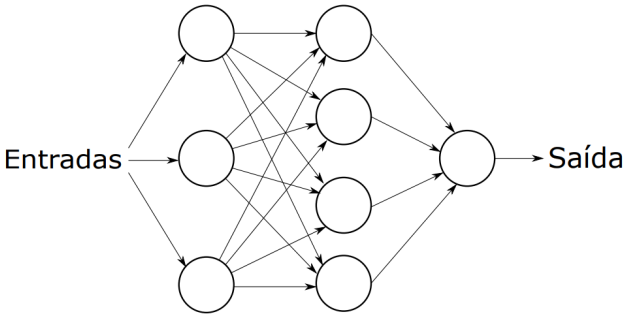


Figura 3 – Exemplo de uma rede MLP.

Cada círculo representa um neurônio. A camada (coluna de neurônios) mais à esquerda nesta rede é chamada de camada de entrada e os neurônios que compõem esta camada são denominados neurônios de entrada. A camada mais à direita, chamada de camada de saída, contém os neurônios de saída. A camada, ou as camadas intermediarias, são as denominadas camadas ocultas, tendo este nome pois elas não são observadas ou monitoradas diretamente como as de entrada ou saída. Essas redes de múltiplas camadas são chamadas de perceptrons multicamadas (MLPs, do inglês *Multilayer perceptrons*). As redes neurais artificiais podem ser classificadas de diferentes formas (Alpaydin 2009).

### 2.2.1 Redes *feedforward* ou recorrentes

Há dois tipos de redes neurais amplamente utilizados, sendo eles denominados *feedforward* ou recorrentes. As redes com arquitetura de tipo *feedforward* são mais comuns em aplicações práticas. Neste caso, a primeira camada é a entrada e a última é a saída, contendo ao menos uma camada oculta. Os processamentos das informações são apenas passadas para frente. Por outro lado, as redes neurais recorrentes possuem ciclos direcionados em seu grafo de conexão, ou seja, seguindo as setas pode se voltar a um neurônio pelo qual o sinal já passou. Elas tem uma dinâmica complicada, tornando-se muito difíceis de treinar.

### 2.2.2 Redes *shallow* ou *deep*

Uma rede neural também pode ser classificada em função da sua complexidade como *shallow*, se possui um número baixo de camadas ocultas. Já uma rede neural é classificada como *deep* quando possui um elevado número de camadas ocultas, o qual permite modelar sistemas muito mais complexos.

### 2.2.3 Redes com conexão densa ou esparsa

Quando cada um dos neurônios de uma camada é conectado à todos os neurônios da camada anterior e posterior, se existirem, essa rede é classificada como densa. Este é o caso mais utilizado para a realização do treino da rede. No caso de quando algumas conexões são deixadas de fora, a rede é classificada como esparsa. É um caso mais específico utilizado quando queremos forçar que a rede consiga resultados sem depender de todas as informações de entrada. Cabe mencionar que em alguns casos, uma rede pode ser inicialmente considerada como densa e apos o treinamento, os enlaces com pesos abaixo de certo limiar são eliminados para reduzir a complexidade da rede na etapa de teste.

### 2.2.4 Redes de regressão ou classificação

As redes neurais podem ser utilizadas para modelar diversos modelos, sendo capazes de retornar um resultado satisfatório. Mas os resultados em cada situação varia muito, e elas podem ser divididas em regressão ou classificação. A classificação é a tarefa de prever um resultado discreto enquanto a regressão prevê resultados contínuos dentro de uma faixa de valores.

## 2.3 APRENDIZADO EM REDES NEURAIS ARTIFICIAIS

O desempenho de uma rede neural artificial dependerá dos valores dos pesos dos enlaces e dos termos de bias, que conjuntamente referem-se como os parâmetros da rede neural. O processo de achar os valores ótimos destes parâmetros denomina-se treinamento, o qual pode ser classificado em três categorias (Alpaydin 2009): O aprendizado supervisionado utiliza conjuntos de dados que inclui valores de entrada e as suas correspondentes saídas. A rede neural é capaz de aprender com esses dados e consegue retornar um resultado desejado, sendo ela de classificação ou regressão. Já no aprendizado não supervisionado dispõe-se dos valores de saída sem ter informação dos valores de entrada associados a estes. Sendo assim, a rede analisa, e aprende por si só padrões e classificações a partir do banco de dados oferecidos. Normalmente aplicado em situações de agrupamento ou associação. Por fim, uma rede neural com aprendizado reforçado aprende por médio de repostas ou *feedbacks*, positivas ou negativas, e usando esse estímulo para realizar o seu treino.

### 2.3.1 Aprendizado supervisionado em redes neurais artificiais perceptron multicamada em configuração de regressão

Considerando uma rede MLP *shallow* em configuração de regressão, o algoritmo de aprendizado mais amplamente utilizado é o denominado retropropagação do erro (14). Para o caso de duas camadas, a propagação do erro será aplicada duas vezes, uma na camada oculta, e uma na camada de saída. A primeira etapa pode ser representada por:

$$z^{(1)} = W^{(1)} \cdot (X^{(1)})^t + b^{(1)} \Rightarrow X^{(2)} = sig(z^{(1)}). \tag{2.4}$$

Por outro lado, a segunda etapa pode ser descrita por:

$$z^{(2)} = W^{(2)} \cdot (X^{(2)})^t + b^{(2)} \Rightarrow \hat{y} = sig(z^{(2)}). \tag{2.5}$$

Desta forma, $z^{(1)}$ pode ser computado multiplicando o vetor peso $W^{(1)}$ da transição entre a camada de entrada e a camada oculta com o vetor dados de entrada $X^{(1)}$, juntamente somado com o vetor bias $b^{(1)}$. O $X^{(2)}$ sendo a aplicação da função de ativação sobre o $z^{(1)}$, representa a saída dos perceptrons da camada oculta. Já $z^{(2)}$ possui uma função similar à $z^{(1)}$, porém sua entrada é justamente a saída da camada anterior. Por fim, o $\hat{y}$ representando a saída da última camada, bem como a saída da rede inteira.

Sabendo como funciona o processo do *feedforward* e como a saída é expressada por um valor ou um vetor de valores, precisamos determinar como quantificar a precisão do nosso modelo. Para isso, será utilizada uma função que compara o resultado da predição $\hat{y}$ com o resultado obtido dos dados do sistema $y$. Esta função é chamada de função de perda ou função de custo (Alpaydin 2009). Este valor é importante para a rede conseguir corrigir a si mesma. Se o resultado da predição com o resultado simulado apresentar uma diferença muito grande em valores, isso representa que a perda é muito larga, ou seja, a precisão da rede é muito baixa. Por outro lado, quanto mais próximo os valores da predição e resultados reais, mais precisa será a rede. Para dados com mais de uma amostra, é feito a média

de perdas, sendo isso considerado a perda empírica. Quando se treina uma rede neural, queremos minimizar a nossa perda empírica entre as predições e os resultados reais.

No caso de regressão, a função de perda mais amplamente utilizada é o erro quadrático médio entre as predições e as saídas do sistema, pois permite obter expressões analíticas para os gradientes desta função. Assim, o termo chamado de perda empírica $J$ pode ser representada como:

$$J = \sum_{i=1}^{m} \frac{1}{2}(\hat{y}_i - y_i)^2, \tag{2.6}$$

sendo $m$ o número de dados de treinamento, $\hat{y}_i$ o valor da saída da rede neural no instante $i$ e $y$ o valor obtido da saída dos dados do sistema no instante $i$. Como utilizar a função de perda de forma com que a rede consiga aprender? A nossa função custo é uma função que tem os pesos como parâmetros. O que precisamos é encontrar os pesos que minimizam a nossa perda $J$. Usaremos $W$ para representar o vetor de pesos entre as camadas, sendo $W^{(1)} = \{w_1^{(1)}, w_2^{(1)}, w_3^{(1)}, ..., w_n^{(1)}\}$ o vetor de pesos entre a camada de entrada e a camada oculta, onde $n$ é o n-ésimo peso, e $W^{(2)} = \{w_1^{(2)}, w_2^{(2)}, w_3^{(2)}, ..., w_m^{(2)}\}$ sendo o vetor de pesos entre a camada oculta e a camada de saída, onde $m$ é o m-ésimo peso.

Derivando a função de custo com relação aos pesos $W$ e os temos de bias, obtemos o gradiente do ponto da função a cada instante, o qual mostra a inclinação máxima naquele ponto, indicando a subida mais íngreme. Como queremos diminuir o valor do custo $J$ afim de aumentar a precisão do modelo, iremos na direção oposta do gradiente. Para computar o gradiente do custo em relação a cada peso $w_n^{(2)}$ na camada de saída, precisamos aplicar a regra da cadeia abrindo a derivada:

$$\frac{\partial J(W)}{\partial w_n^{(2)}} = \frac{\partial J(W)}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_n^{(2)}}. \tag{2.7}$$

Para obter o gradiente do custo em relação a cada elemento de $w_n^{(1)}$ na camada oculta, também aplicaremos a regra da cadeia:

$$\frac{\partial J(W)}{\partial w_n^{(1)}} = \frac{\partial J(W)}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z_1} \cdot \frac{\partial z_1}{\partial w_n^{(1)}}. \tag{2.8}$$

Desta forma, calculando os gradiente para os sucessivos pesos, conseguimos realizar o *backpropagation* do erro reutilizando informações obtidas na saída para fazer uma atualização no valor dos pesos $W$:

$$W_{k+1} = W_k - \alpha \frac{\partial J(W_k)}{\partial W_k}. \tag{2.9}$$

Na expressão anterior, $\alpha$ representa a taxa de aprendizado, a qual controla o passo com que atualizamos os pesos a cada iteração. Lembrando que o gradiente nos diz a direção que deve se tomar, mas não a distância (Alpaydin 2009). Selecionar um valor de $\alpha$ correto não e trivial, pois se a taxa de aprendizado for muito pequena, o modelo convergirá muito devagar poderia ficar preso num mínimo local, não podendo escapar dele. Por outro lado, se a taxa de aprendizado for alta demais, poderia não achar o mínimo é incluso ficar instável. A Figura 4 mostra graficamente o efeito de utilizar uma taxa de aprendizado $\alpha$ excessivamente baixa (a) e alta (b). Um método para evitar ou, ao menos, minimizar,

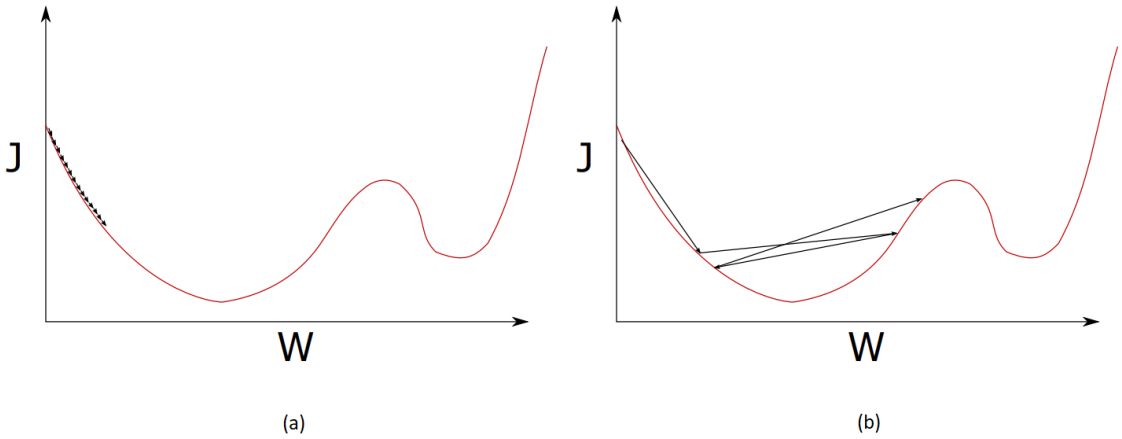


Figura 4 – Influência de $\alpha$ no treino. (a) $\alpha$ muito pequeno, aumenta significantemente o tempo para convergir, podendo ficar preso em locais mínimos. (b) $\alpha$ Muito grande, acontecerá a ultrapassagem do mínimo, podendo nunca convergir.

o efeito deste problema é ter uma taxa de aprendizado adaptativo, onde pode aumentar ou diminuir dependendo de quão largo o gradiente é (Alpaydin 2009). Podemos resumir o algoritmo de treino da seguinte forma:

1. Inicializar os pesos e bias aleatoriamente;
2. Aplicar o processo de *feedforward*;
3. Calcular o custo $J$;
4. Computar o gradiente para cada parâmetro;
5. Atualizar os valores dos parâmetros;
6. Repetir os passos 2-5 até o custo $J$ convergir.

Cada processo de atualização de pesos realizado é chamado de iteração. Dependendo da complexidade do problema, da quantidade de dados e do modelo de rede, o tempo do treino pode variar significativamente. Ao finalizar o treinamento, conseguimos encontrar os valores de pesos $w$ e bias $b$ para ser armazenado, e utilizar dados não incluídos no treino para aplicar um processo de *feedforward*. Com isso, é possível quantificar a capacidade do sistema para predizer o comportamento do sistema em presencia de novos dados de entrada.

De forma geral, este é o jeito de treinar uma rede neural. Porém na prática a função de custo pode ter múltiplos mínimos locais, tal e como mostrado na Figura 5, tornando o processo muito mais complexo e complicado, e requerendo métodos de optimização mais sofisticados para um funcionamento mais satisfatório. Uma das maiores dificuldades no processo de treinamento é que o método com gradiente sempre leva para o mínimo local mais próximo, podendo não convergir para o mínimo global. Por tanto, atingir o mínimo global pode ser muito desafiador. De fato, a rede pode convergir a diferentes mínimos locais dependendo da inicialização dos pesos no início do treino.

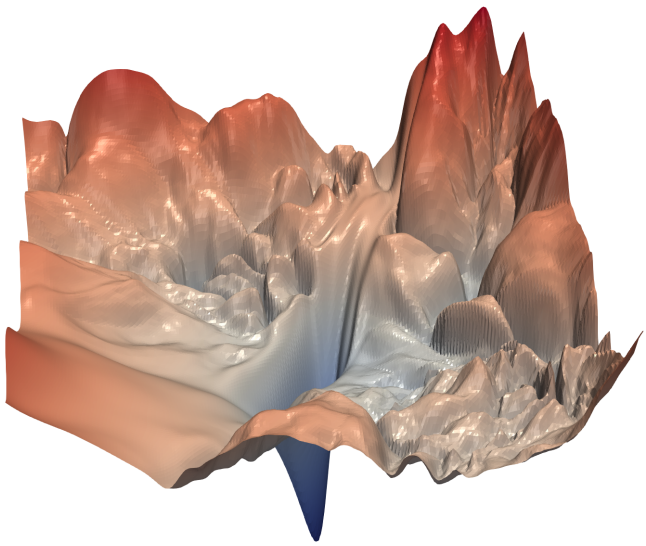

Figura 5 – Exemplo de um panorama de uma função de perda (Li et al. 2017)

.

### 2.3.2 *Overfitting* e *biasing*

Idealmente queremos um modelo que consegue predizer com a maior precisão possível o comportamento do sistema (saídas) na presencia de novos dados. Para isso, o modelo deve ser capaz de extrair informação do sistema e filtrar o ruído que afeita os dados de treinamento. Porém, quando a rede neural tem um número elevado de graus de liberdade, a rede começa aprender caraterísticas do ruído que por estar descorrelacionado com o ruído que afeita aos dados de teste, causa uma penalização na capacidade de predizer novas saídas. Este efeito é denominado *overfitting*. Existem diferentes alternativas para minimizar o impacto do *overfitting* na precisão da rede neural, a primeira é a redução do número de graus de liberdade, por exemplo, utilizando um menor número de neurônios. A segunda alternativa é a introdução de termos de regularização que penalizam a função de custo. Por último, podemos evitar o efeito de *overfitting* detendo o processo de treinamento antes de que o modelo comece a aprender as caraterísticas do ruído. Por outro lado, em caso de não permitir que o modelo aprenda as caraterísticas básicas do sistema, devido a um número de graus de liberdade baixo ou por um processo de treinamento insuficiente, o sistema apresenta o efeito denominado *biasing* ou *underfitting*. A Figura Figura 6 mostra qualitativamente o efeito de *undefitting*, fitado ideal, e *overfitting*.

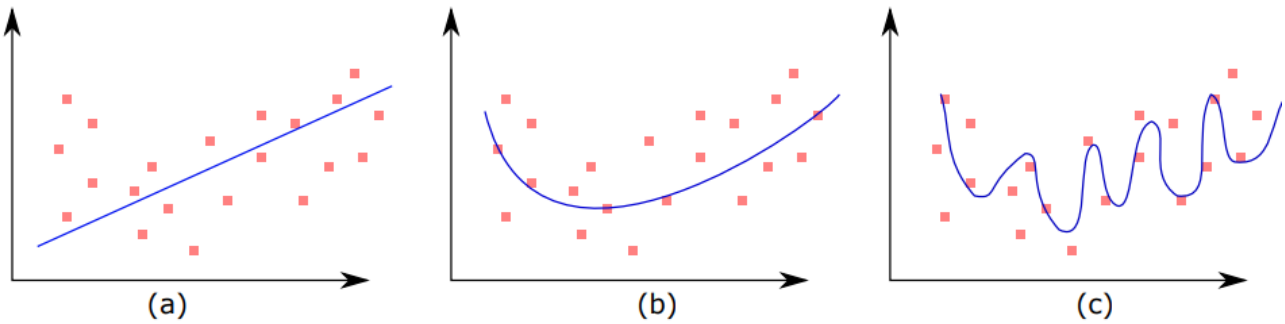


Figura 6 – Diferentes modelados do sistema: (a) modelado sob efeito de *underfitting*, (b) modelado com fitado ideal, e (c) modelado com problema de *overfitting*.

## 2.4 RECAPITULAÇÃO

Para fazer uma recapitulação geral do funcionamento de uma rede neural, podemos identificar três tópicos principais, os quais são esquematizados na Figura 7.

- Perceptron
  - São blocos de construção estrutural com capacidade de processar operações matemáticas.
  - Possuem funções de ativação não linear.
- Rede neural
  - Formado por combinação de perceptrons em diferentes tipos de arquitetura.
  - Treinamento feito por *feedforward* em problema de regressão.
  - Otimização feito por meio de *backpropagation*.
- Treino em prática.
  - Problema de taxa de aprendizagem
  - Problema de *overfitting*

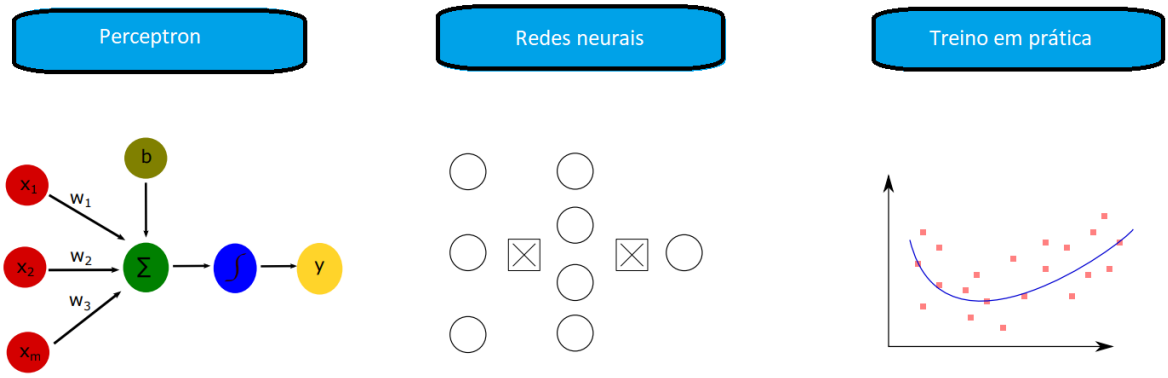


Figura 7 – Representação gráfica da recapitulação.

# 3 SIMULAÇÃO DO EDFA E MODELO DE EDFA BASEADO EM REDES NEURAIS ARTIFICIAIS

## 3.1 SIMULAÇÃO DO EDFA

A fim de obter dados precisos do comportamento de um EDFA, utilizou-se o simulador VPI Transmission Maker que permite simular fibras dopadas com érbio. Especificamente, simulamos um amplificador de dois etapas. A primeira delas é composta por uma secção de fibra dopada de 20 m com bombeio óptico a 980 nm co-propagante. A segunda etapa, está composta por outra secção de fibra dopada mas, neste caso, com bombeio contra-propagante a 1490 nm. Consideraram-se 4 canais WDM em torno de 1550 nm com separação de 200 GHz [1]. Em total, 4500 diferentes configurações de potência dos canais de entrada foram simuladas. As potências de entrada foram uniformemente distribuídas entre -20 dBm e -10 dBm.

## 3.2 ARQUITETURA ADOTADA

O primeiro passo para desenvolver o modelo de um EDFA baseado em redes neurais é a identificação das variáveis de entrada e de saída. No nosso caso, estamos interessados em predizer o ganho em sistemas WDM e, por tanto, as nossas variáveis serão as potências dos canais de entrada e de saída. Como as potências de saída são variáveis contínuas, a rede neural operará em configuração de regressão. Por outro lado, podemos considerar uma rede de múltiplas saídas capaz de predizer as potências de saída de todas as saídas simultaneamente ou, alternativamente, implementar uma rede dedicada para cada saída. Decidiu-se pela segunda opção pois esta permite uma adaptação maior dos pesos a cada uma das saídas. Cabe destacar que mesmo considerando cada saída separadamente, a interação entre as diferentes entradas continua sendo considerada. Desta forma, implementamos múltiplas redes neurais de múltiplas entradas e única saída (MISO, do inglês *multiple inputs single output*) em paralelo. Em relação à função de custo, como já indicado, o erro quadrático médio foi considerado. Porém, cabe mencionar que devido à grande faixa dinâmica das potências de entradas, o erro quadrático médio foi calculado considerando as potências em dBm, o qual assegura que se minimiza o erro quadrático médio relativo.

Na seleção da topologia devemos considerar que o sistema não apresenta memória e, por conseguinte, é mais fatível considerar uma rede *feedforward*. Por outro lado, considerando que o sistema não é muito complexo, será adotada uma rede *shallow*. Finalmente, tendo em conta que todas as entradas *a priori* tem um peso similar na saída, utilizará-se uma rede densa. Assim, a rede neural será uma MLP MISO com uma única camada oculta densamente conetada operando em configuração de regressão. Em quanto ao número de neurônios, a camada de entrada terrá 4 neurônios, correspondendo ao número de canais WDM considerados, a camada de saída terrá um único neurônio e a camada oculta terra um número de neurônios que será otimizado numericamente.

[1] Utilizamos um número pequeno de canais WDM para poder entender os fundamentos das redes neurais, porém prevemos considerar um numero maior de canais no futuro. Por outro lado, se decidiu usar uma separação de 200 GHz a fim de ter em conta a dependência frequencial do ganho do amplificador.

## 3.3 PREPROCESSAMENTO DOS DADOS E TREINAMENTO DO MODELO

Os dados obtidos por simulação devem ser processados antes de poder ser utilizados para treinar e validar o modelo. Em particular, é importante normalizar os dados para que fiquem limitados ao intervalo [-1,1]. Uma vez normalizados, os dados são divididos em dois conjuntos: 70% das combinações foram utilizadas como conjunto de treinamento enquanto que 15% das combinações foram utilizadas para a validação cruzada, e o resto das combinações foram utilizadas para teste a fim de validar o modelo. Por outro lado, como já mencionado, a capacidade do modelo para predizer a potência de saída diante novas combinações de potências de entrada depende criticamente dos valores dos pesos $W$ que por sua vez dependem dos valores adotados na inicialização aleatória dos pesos. Para minimizar sensibilidade da inicialização, cada configuração foi otimizada 50 vezes com valores inciais diferentes e o valor médio foi calculado.

# 4 RESULTADOS

Como mencionado no capítulo anterior, a função de custo considerada para avaliar a procissão do modelo é o erro quadrático médio entre a saída do sistema (simulação realizadas em VPI Transmission Maker) e o modelo baseado em redes neurais artificiais. Além disso, cabe relembrar que implementamos uma rede neural para cada canal de saída. Desta forma, para cada saída mostramos a função de custo tanto do traino como de teste (validação cruzada) em função do número de neurônios e do número de iterações para o número de neurônios ótimo. Também representamos os valores previstos em função dos valores simulados para o conjunto de treinamento e do validação cruzada.

No caso da primeira saída, podemos observar que quando a função de custo é representada em termos do número de neurônios da camada oculta, Figura 8(a), a função de custo tanto do treinamento como de validação reduzem-se a medida que o número de neurônios aumenta até os 7 neurônios. Este efeito é facilmente entendível pois o modelo progressivamente aumenta a complexidade, sendo capaz de modelar com maior precisão. Para um número de neurônios entre 7 e 15, as funções de custo permanecem relativamente estáveis, enquanto que para números de neurônios maiores, as funções de custo deterioram-se, o qual pode ser atribuído a um treinamento menos eficiente pois ao aumentar o número de neurônios é mais provável convergir para um mínimo local. Em quanto à evolução da função de custo em termos do número de iterações para uma camada oculta de 7 neurônios mostrado na Figura 8(b), pode se observar que efetivamente decaem monotonamente. Neste ponto, é importante destacar que a maior separação entre as funções de custo do conjunto de treinamento e de validação pode confundir devido à representação logarítmica. Porém, em termos absolutos a diferença entre as duas funções de custo é reduzida, mostrando que efetivamente não acontece *overfitting*, pois em caso de este acontecer, as curvas de custo de do set de treino e da validação cruzada detergeriam. Outro ponto importante a observar é que a função de custo atinge um valor relativamente estável após aproximadamente 50,000 iterações. Já na representação dos valores previstos (saída da rede neural) em função dos simulados (saída do simulador VPI Transmission Maker) mostradas nas Figura 9(a) e Figura 9(b), podemos concluir que mesmo apresentando um pequeno desvio, o modelo consegue predizer a saída.

O comportamento é similar para o resto de saídas. As Figura 10 e Figura 11 para a saída 2, as Figura 12 e Figura 13 para a saída 3 e as Figura 14 e Figura 15 para a saída 4. Comparando os resultados, podemos observar que o erro quadrático médio mínimo no conjunto de treinamento é em torno a $10^{-3}$ dB para as saídas 1 e 4, enquanto que este valor é menor para as saídas 2 e 3. A maior precisão na predição das saídas 2 e 3 também pode ser apreciada nas gráficas que mostram o valor predito pelo modelo em função do valor simulado. Em particular, pode se observar como para as saídas 2 e 3, o valor predito é mais próximo ao desejado. Este comportamento em que o modelo apresenta uma precisão menor nos valores extremos é comum a muitos problemas de regressão.

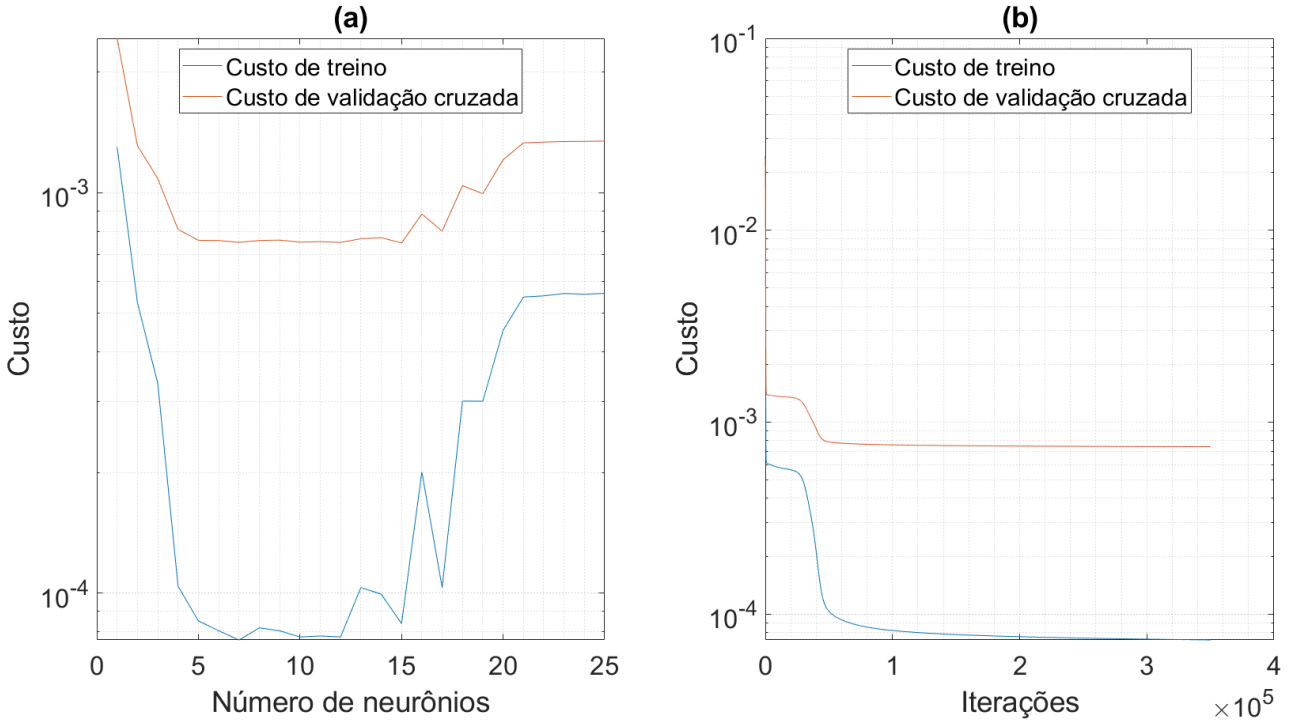


Figura 8 – Resultado da rede construída para o primeiro canal do sistema WDM: (a) curva de custo em função do número de neurônios na camada oculta e (b) curva de custo em termos do número de iterações de treino.

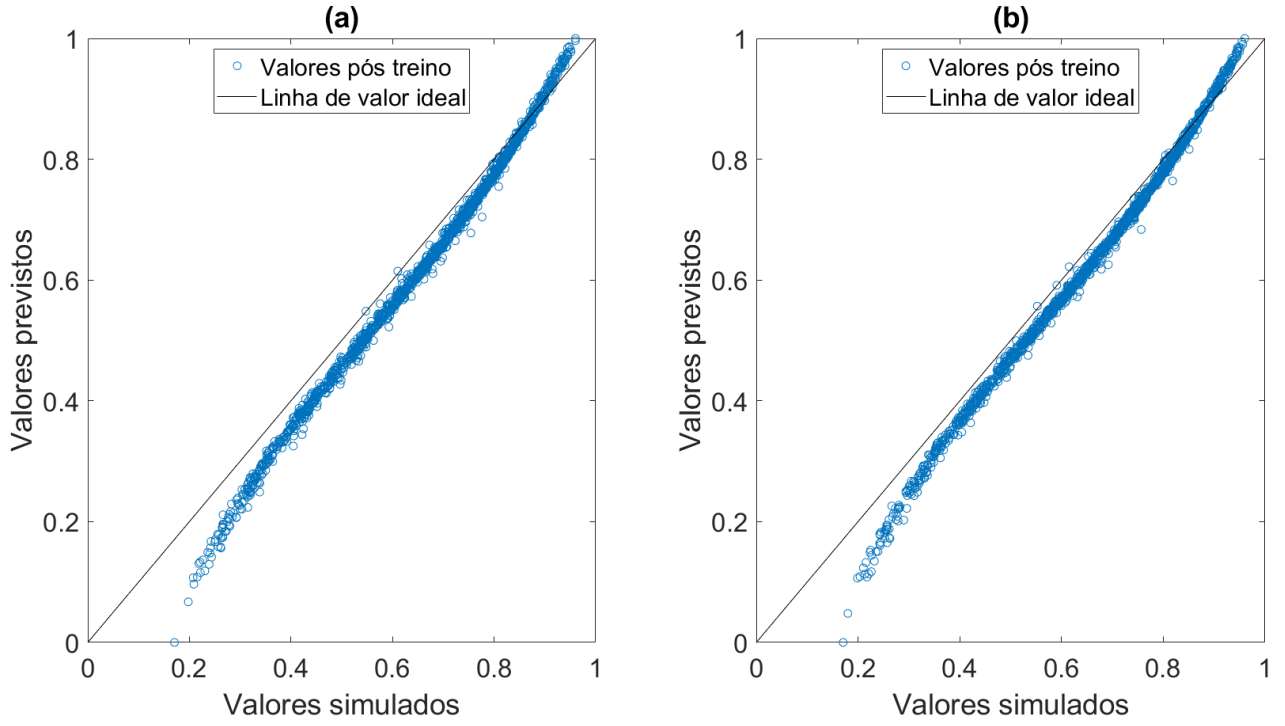


Figura 9 – Avaliação da precisão do modelo para o primeiro canal do sistema WDM. Comparação de valores de saída do modelo e do sistema para os dados de (a) treino e (b) teste.

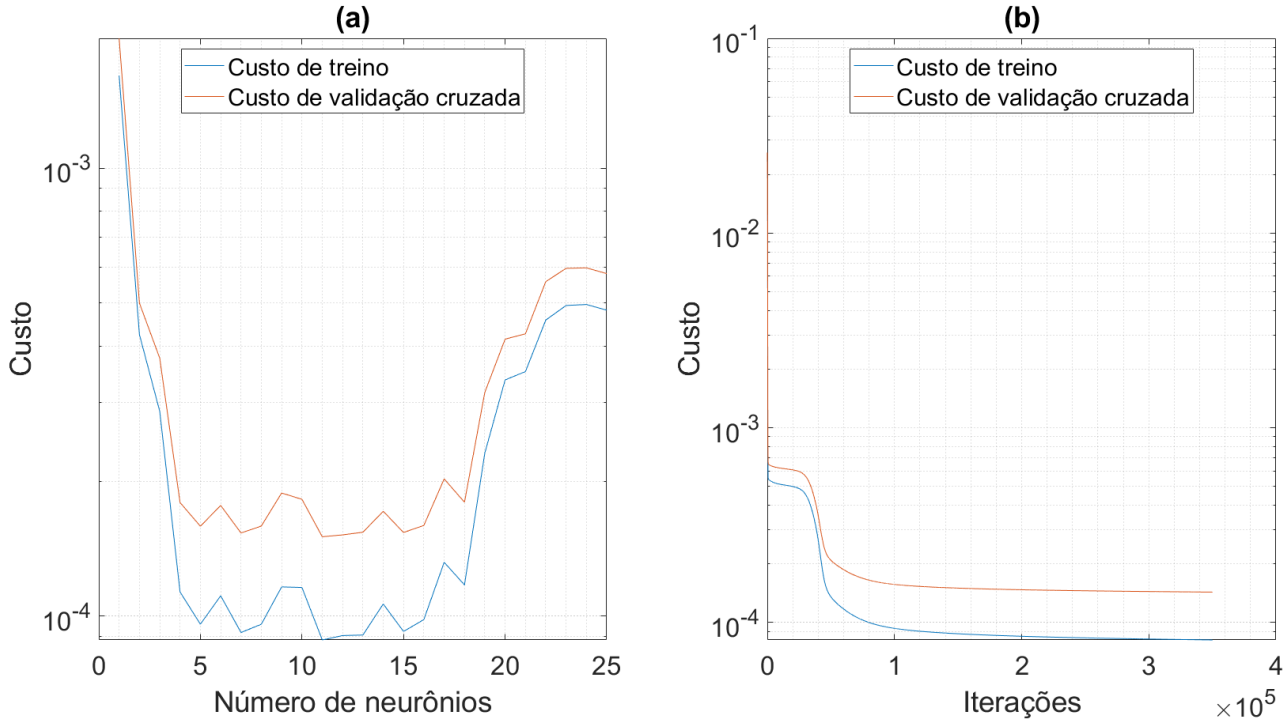


Figura 10 – Resultado da rede construída para o primeiro canal do sistema WDM: (a) curva de custo em função do número de neurônios na camada oculta e (b) curva de custo em termos do número de iterações de treino.

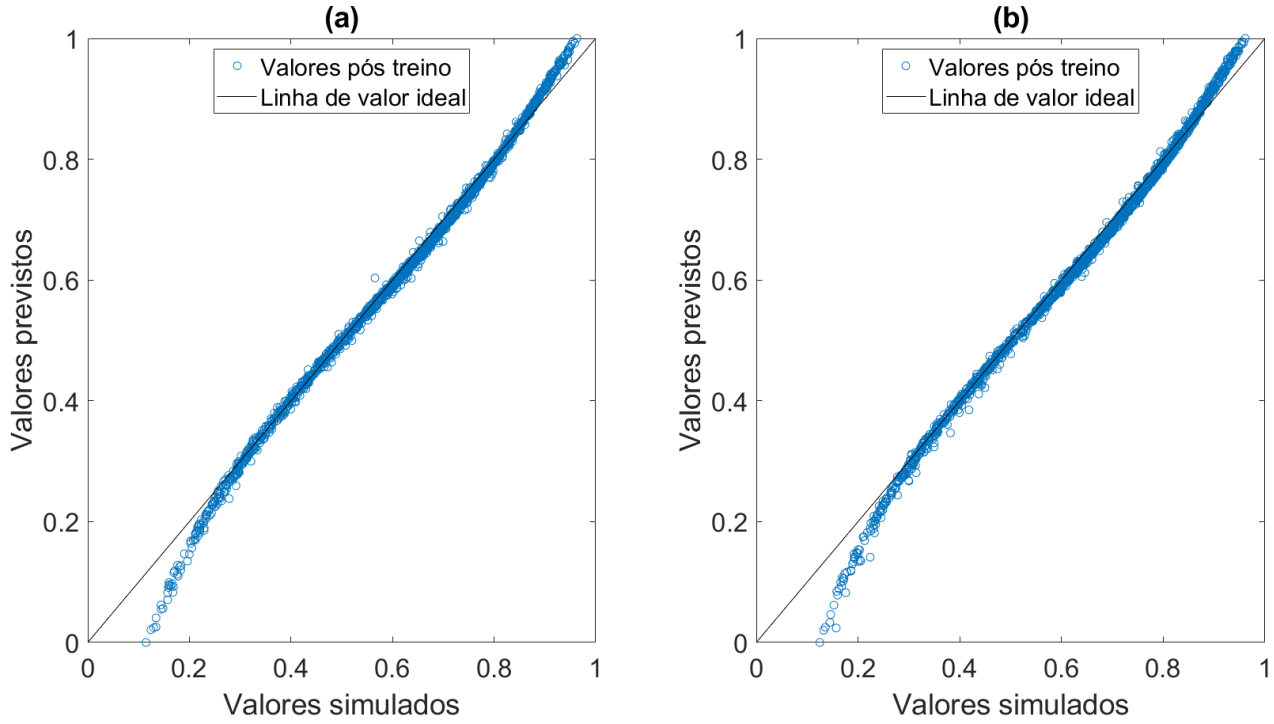


Figura 11 – Avaliação da precisão do modelo para o primeiro canal do sistema WDM. Comparação de valores de saída do modelo e do sistema para os dados de (a) treino e (b) teste.

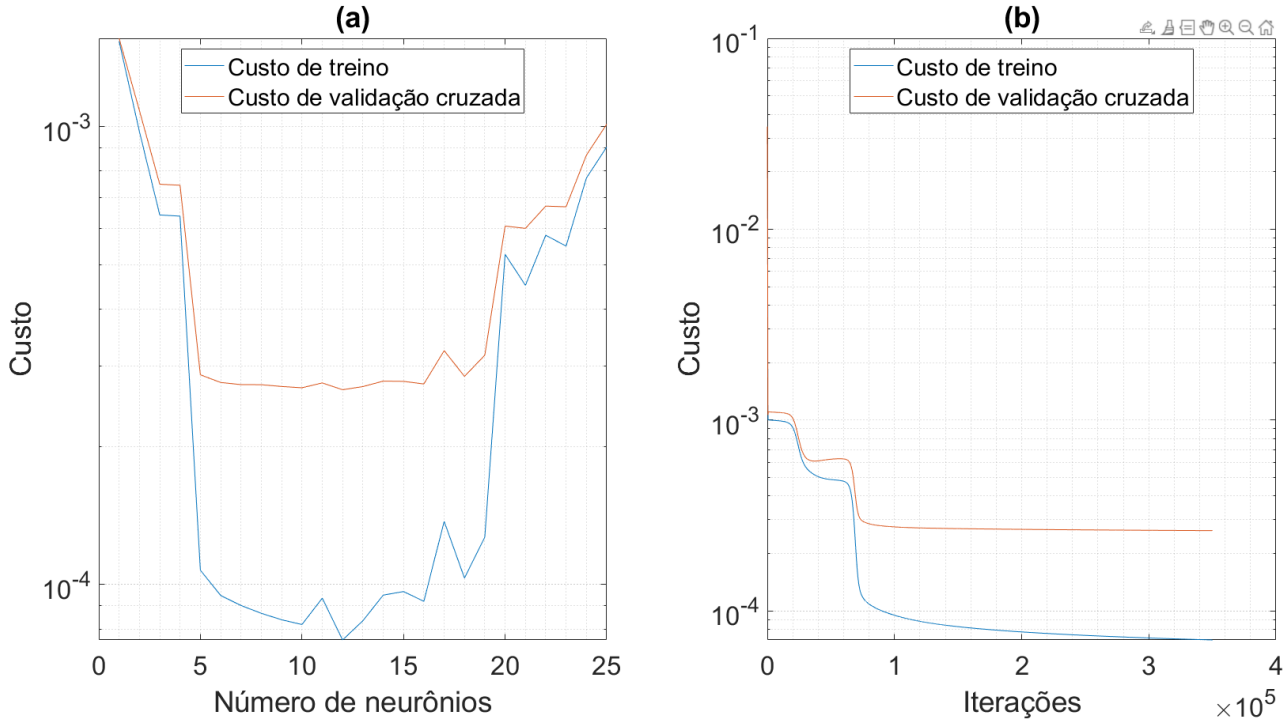


Figura 12 – Resultado da rede construída para o primeiro canal do sistema WDM: (a) curva de custo em função do número de neurônios na camada oculta e (b) curva de custo em termos do número de iterações de treino.

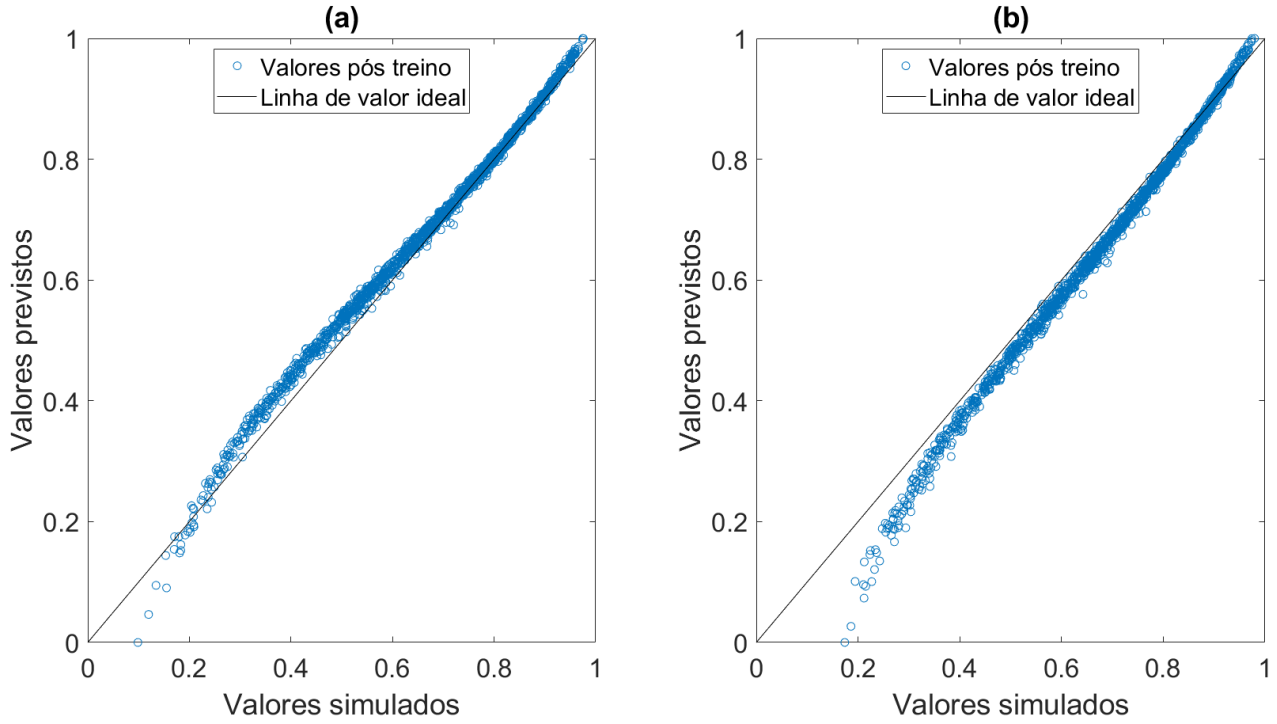


Figura 13 – Avaliação da precisão do modelo para o primeiro canal do sistema WDM. Comparação de valores de saída do modelo e do sistema para os dados de (a) treino e (b) teste.

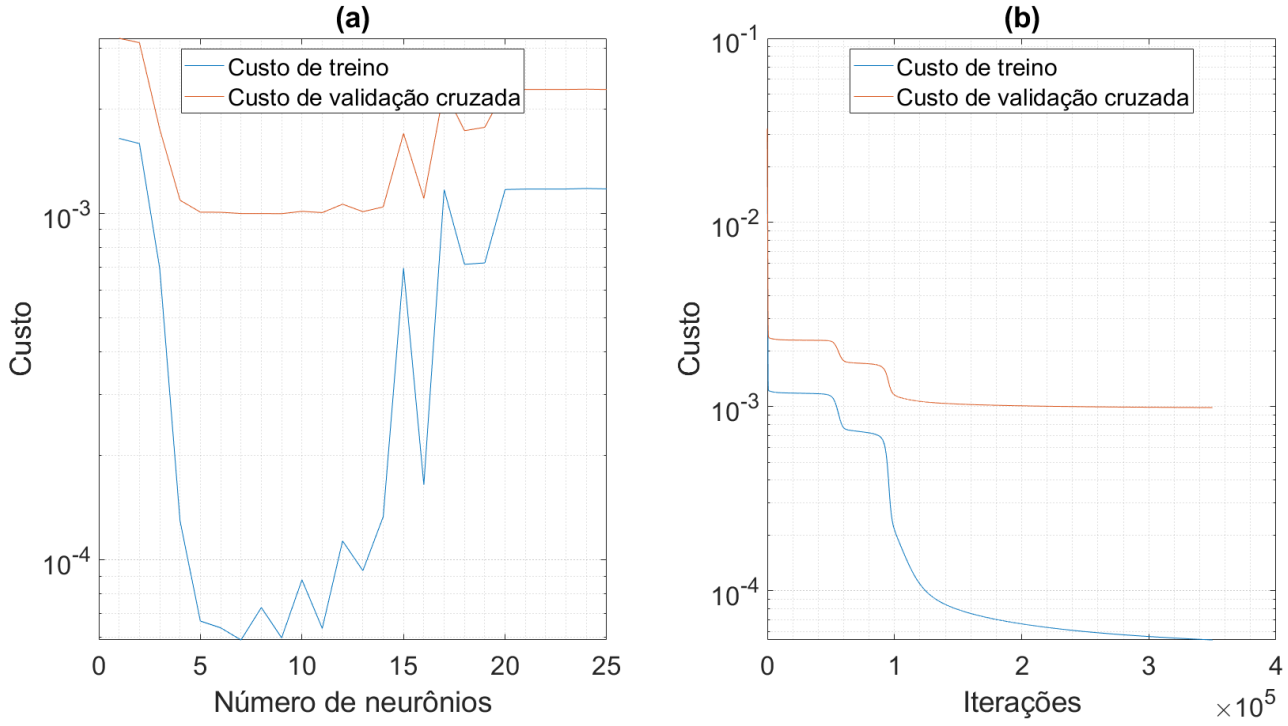


Figura 14 – Resultado da rede construída para o primeiro canal do sistema WDM: (a) curva de custo em função do número de neurônios na camada oculta e (b) curva de custo em termos do número de iterações de treino.

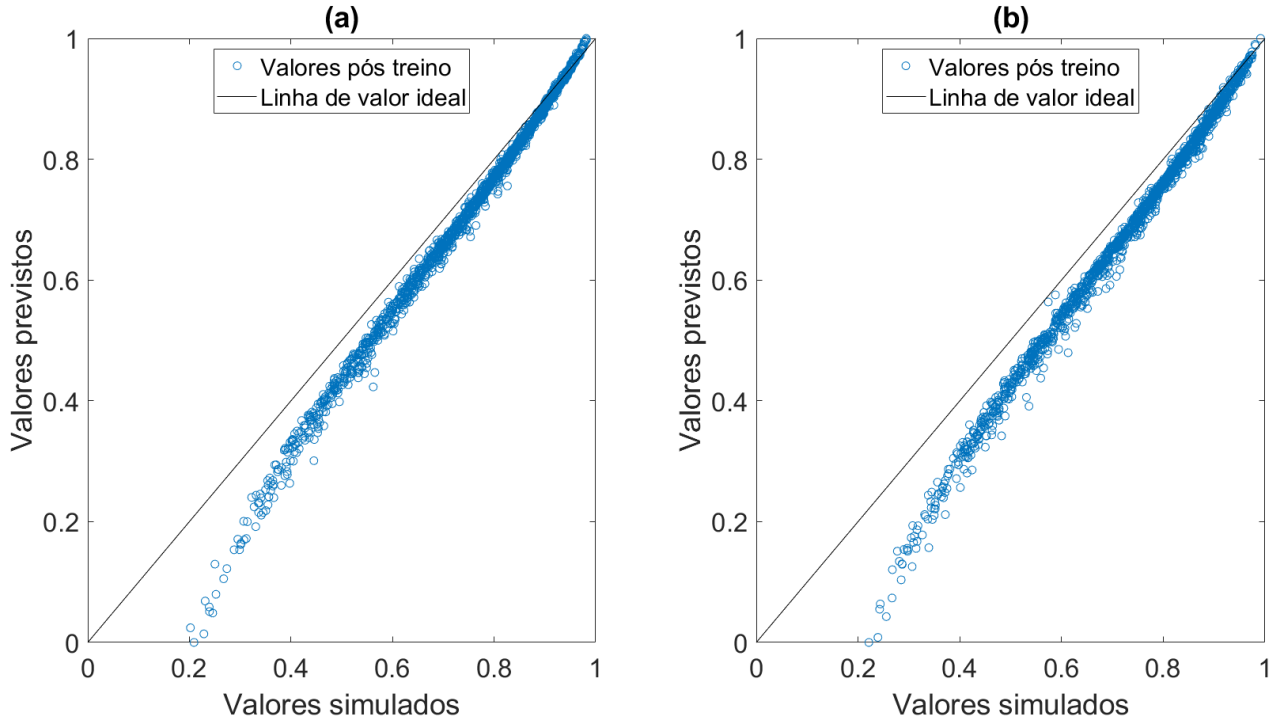


Figura 15 – Avaliação da precisão do modelo para o primeiro canal do sistema WDM. Comparação de valores de saída do modelo e do sistema para os dados de (a) treino e (b) teste.

Nesse estudo foi feito treinamento para cada saída do amplificador. Para que seja possível identificar qual o número de neurônios adequados na camada oculta para que a rede consiga representar melhor o

grupo de dados, foi feito uma variação no número de neurônios entre 1 e 25. Então o número com o menor custo será a utilizada para realizar a comparação dos valores preditos e dos valores simulados. Como pode ser visto à esquerda, nas Figuras 8, 10, 12 e 14, o número de neurônios na camada oculta em em treinos para cada saída de forma que tenha o menor custo, são respectivamente, 7, 11, 12 e 7. Foi traçado também a curva de custo por iteração, mostrando a convergência do custo ao longo da curva. Vale a pena notar que, para esse estudo, embora a curva da validação cruzada aparenta estar ficando mais distante da curva de treinamento, este não é o caso, visto que o gráfico está na escala logarítmica. Nas Figuras 9, 12, 13 e 15, podemos ver a comparação dos valores previstos com os valores simulados. À esquerda utilizando os dados de treino, e à direita usando os dados de teste e os parâmetros da rede para verificar a precisão.

## 4.1 ANÁLISE DA COMPLEXIDADE COMPUTACIONAL

Por outro lado, é possível analisar a complexidade das redes neurais artificiais construídas neste estudo computando a quantidade de operações de ponto flutuante realizadas após o treinamento da rede, na etapa de validação cruzada. Observando a Equação (2.3), o número de operações em cada neurônio é igual a duas vezes (considerando o número de multiplicações e o número de somas) o número de neurônios na camada prévia. Neste cálculo não é considerado o custo computacional da função de ativação pois pode-se assumir que está é implementada utilizando tabelas de busca. Portanto, se considerarmos o número de neurônios na camada de entrada $N_i$, o número de neurônios na camada oculta de $N_h$, e o número de neurônios na camada de saída como $N_o$, poderemos calcular o número de operações de ponto flutuante para cada rede dedicada, $N_{op}$, utilizando a formula a seguir:

$$N_{op} = 2N_iN_h + 2N_hN_o = 2N_h \cdot (N_i + N_o). \tag{4.1}$$

Aplicando a Equação (4.1) para cada uma das saídas do amplificador simulado e, considerando o número ótimo de neurônios para a camada oculta obtido na analise da subsecção anterior, obtemos o número de operações de 70, 110, 120 e 70, totalizando 370 operações de ponto flutuante. A Tabela 2 resume os resultados para as diferentes saídas, assim como o número total de operações.

Tabela 2 – Número de operações para cada canal, considerando o número de neurônios ótimo na camada oculta.

| | Número de neurônios | | | Número de |
|---|---|---|---|---|
| Saída | Camada de entrada ($N_i$) | Camada oculta ($N_h$) | Camada de saída ($N_o$) | operações ($N_{op}$) |
| 1 | 4 | 7 | 1 | 70 |
| 2 | 4 | 11 | 1 | 110 |
| 3 | 4 | 12 | 1 | 120 |
| 4 | 4 | 7 | 1 | 70 |
| | | | Total | 370 |

# 5 CONSIDERAÇÕES FINAIS

Neste trabalho o efeito de modulação de ganho cruzada em amplificadores EDFAs em sistemas WDM foi modelado utilizando redes neurais artificiais. Os resultados mostram que efetivamente as redes neurais mostradas (MLPs *feedforward* densamente conetados) permitem modelar a potência de saída dos canais mesmo quando estes canais estão afetados por modulação de ganho cruzada. O estudo também mostra que os canais centrais requerem de um maior número de neurônios na camada oculta, isto é, um modelo mais complexo.

Este trabalho constitui um primeiro passo para entender o modelo de amplificadores EDFA utilizando redes neurais artificiais e estabelece as bases para trabalhos futuros. Em particular, podemos mencionar as seguintes aplicações e aprimoramentos:

- Considerar um maior número de canais: É importante considerar um maior número de canais pois em cenários reais é previsível que o número de canais seja muito maior. Este aumento de canais, evidentemente aumentará a complexidade do sistema e por tanto afeitará ao número de neurônios (e pode ser o número de camadas ocultas também).

- Considerar a razão sinal-ruído (SNR): Além do ganho, a relação sinal a ruído também depende das potências e das frequências dos canais. Seria interessante implementar uma (ou várias) redes neurais paralelas para predizer não somente a potência de saída mas também o nível de ruído de cada canal na saída do amplificador.

- Considerar as potências de bombeio: Neste trabalho, as potências de bombeio não foram variadas. Porém, é claro que estas tem um efeito significativo tanto no ganho como no ruído do amplificador.

- Configuração do EDFA para potência plana/SNR plana: Uma vez modelo o EDFA, é possível aplicar algoritmos de optimização para atingir potências de saída (ou SNRs) o mais planas possíveis.

# REFERÊNCIAS


AGRAWAL, G. P. **Fiber-optic communication systems**. [S.l.]: John Wiley & Sons, 2012. v. 222.

AGRAWAL, G. P. Optical communication: its history and recent progress. In: **Optics in our time**. [S.l.]: Springer, Cham, 2016. p. 177–199.

ALPAYDIN, E. **Introduction to machine learning**. [S.l.]: MIT press, 2009.

ANTHONY, M.; BARTLETT, P. L. **Neural network learning: Theoretical foundations**. [S.l.]: cambridge university press, 2009.

BAUMGARTNER, R.; BYER, R. Optical parametric amplification. **IEEE Journal of Quantum Electronics**, IEEE, v. 15, n. 6, p. 432–444, 1979.

BROMAGE, J. Raman amplification for fiber communications systems. **journal of lightwave technology**, IEEE, v. 22, n. 1, p. 79, 2004.

DESURVIRE, E.; ZERVAS, M. N. Erbium-doped fiber amplifiers: principles and applications. **Physics Today**, v. 48, n. 2, p. 56, 1995.

DUTTA, N. K.; WANG, Q. **Semiconductor optical amplifiers**. [S.l.]: World scientific, 2013.

GILES, C. R.; DESURVIRE, E. Modeling erbium-doped fiber amplifiers. **Journal of lightwave technology**, IEEE, v. 9, n. 2, p. 271–283, 1991.

LI, H. et al. Visualizing the loss landscape of neural nets. **arXiv preprint arXiv:1712.09913**, 2017.

MARHIC, M. E. **Fiber optical parametric amplifiers, oscillators and related devices**. [S.l.]: Cambridge university press, 2008.

MEARS, R. The edfa: past, present and future. In: IEEE. **Fifth Asia-Pacific Conference on... and Fourth Optoelectronics and Communications Conference on Communications,**. [S.l.], 1999. v. 2, p. 1332–vol.

MENIF, M. et al. Cross-gain modulation effect on the behaviour of packetized cascaded edfas. **Journal of Optics A: Pure and Applied Optics**, IOP Publishing, v. 3, n. 3, p. 210, 2001.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. **nature**, Nature Publishing Group, v. 323, n. 6088, p. 533–536, 1986.

SINGH, S.; SINGH, A.; KALER, R. Performance evaluation of edfa, raman and soa optical amplifier for wdm systems. **Optik**, Elsevier, v. 124, n. 2, p. 95–101, 2013.

WAGNER, C. et al. Wavelength-agnostic wdm-pon system. In: IEEE. **2016 18th International Conference on Transparent Optical Networks (ICTON)**. [S.l.], 2016. p. 1–4.

WANG, S.-C. Artificial neural network. In: **Interdisciplinary computing in java programming**. [S.l.]: Springer, 2003. p. 81–100.

ZIMMERMAN, D. R.; SPIEKMAN, L. H. Amplifiers for the masses: Edfa, edwa, and soa amplets for metro and access applications. **Journal of lightwave technology**, IEEE, v. 22, n. 1, p. 63, 2004.